
B8IT108 Data and Web Mining CA

Data and Web Mining CA – Report

Ciaran Finnegan / Dermot Madsen

Student No : 10524150 / 10522567

9/02/2020

List of contents

1	Project Overview	3
1.1	High Level Description.....	3
1.2	The CRISP-DM Methodology / Reference Model	3
1.3	Development Environments.....	6
2	Business Understanding	7
2.1	Determine Business Objectives	7
2.2	Assess Situation	8
2.3	Determine Data Mining Goals.....	9
2.4	Produce Project Plan.....	9
3	Data Understanding	11
3.1	Collect Initial Data.....	11
3.2	Describe Data.....	12
3.3	Explore Data.....	13
3.4	Verify Data Quality.....	22
4	Data Preparation	24
4.1	Select Data.....	24
4.2	Clean Data	27
4.3	Construct Data	28
4.4	Integrate Data.....	39
4.5	Format Data	39
5	Modelling	40
5.1	Select Modelling Technique.....	40
5.2	Generate Test Design	42
5.3	Build Model.....	43
5.4	Assess Model	53
6	Evaluation.....	55
6.1	Evaluate Results	55
6.2	Review Process.....	55
6.3	Determine Next Steps	56
7	Deployment	57
7.1	Plan Deployment	57
7.2	Plan Monitoring and Maintenance	57
7.3	Produce Final Report.....	58
7.4	Review Project	58
8	Appendices and References	59
8.1	Appendix A: Understanding Wine and Types.....	59
8.2	Appendix B: The White Wine Dataset.....	61
8.3	Appendix C: References.....	64



1 Project Overview

1.1 High Level Description

This document covers our planned approach and execution of a data mining analysis on a dataset relating to the assessment and prediction of wine quality.

Following the CRISP-DM model, we laid out an objective for this Continuous Assessment exercise and followed a series of steps, often iteratively, to arrive at a predictive model for wine quality based on known feature characteristics.

The following sections explain the business objectives, the assessment of data, and the selection, implementation and deployment of a model to provide a predictive guide to new wine quality.

1.2 The CRISP-DM Methodology / Reference Model

In the mid to late 1990s, business markets were showing a sharp upturn in interest into the possibilities offered by data mining practices. The need for a standard process model, widely and freely available, became quickly apparent.

By 1999/2000, a process model named CRISP-DM (Cross-Industry Standard Process for Data Mining) had been produced by leading thinkers in the industry. It was based on practical, real-world experiences and sought input across a range of business domains.

As explained in the following sections of this document, we have taken the key principles of CRISP-DM to implement our CA project.

1.2.1 Methodology

The CRISP-DM methodology is described as a hierarchical process mode with four levels that transition from the generic to the specific;

1. **Phases** – process blocks consisting of several generic tasks.
2. **Generic Tasks** – so called because they are intended to be robust and stable tasks that can apply in any data mining situation.
3. **Specialised Tasks** – a description as to how the generic tasks should be applied in specific situations. Very often these tasks can be performed in multiple orders and repeated a number of times.
4. **Process Instances** – this is a record of the actions, decisions, and results of an actual data mining engagement.



1.2.2 Reference Model

The life cycle of a data mining project consists of six phases, as shown in this image below.

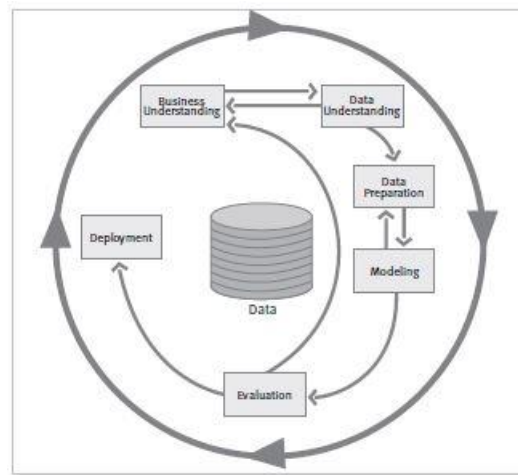


Figure 1

The sequence of the phases is not rigid. In our Wine Quality project moving back and forth between different phases was frequently required, as expected.

The outcome of each phase determines which phase, or particular task of a phase, has to be performed next. The arrows indicate the most important and frequent dependencies between phases.

As an example, in our Wine Quality CA we needed to iterate between various data balancing options in the Data Preparation Phase and move back and forth between the Modelling Phase.

The diagram above displays the following phases.

- Business Understanding
- Data Understanding
- Data Preparation
- Modelling
- Evaluation
- Deployment

The next sections of this document elaborate on these phases a little further and the remainder of the report describes the actual implementation of the CRISP-DM against our Wine Quality project.



1.2.3 Business Understanding

Understand the project objectives and the requirements from a business perspective.

Convert this knowledge into an actual data mining problem definition, along with a project plan that will provide a framework to deliver the business objectives.

1.2.4 Data Understanding

Start with initial data collection.

Proceed into activities that provide familiarity with the data, including data quality issues, data insight, and possible sub-sets within the data.

1.2.5 Data Preparation

Activities to construct the final dataset that will be fed into the modelling too.

Data preparation tasks can be performed in multiple orders and over many interactions.

1.2.6 Modelling

Select and apply various modelling techniques.

Calibrate parameters to provide optimal values within the model.

Revert to data preparation phase, if necessary.

1.2.7 Evaluation

A high quality data analysis model has been built.

Assess that the model achieves the business objective.

1.2.8 Deployment

The knowledge gained by the creation of the model will need to be organised and presented in a way that it can be used by the customer.

It is important for the customer to know up front what actions need to be carried out in order to actually make use of the created models.



1.3 Development Environments

The majority of development took place within the RapidMiner toolkit and screenshots are provided to show the step-by-step working.

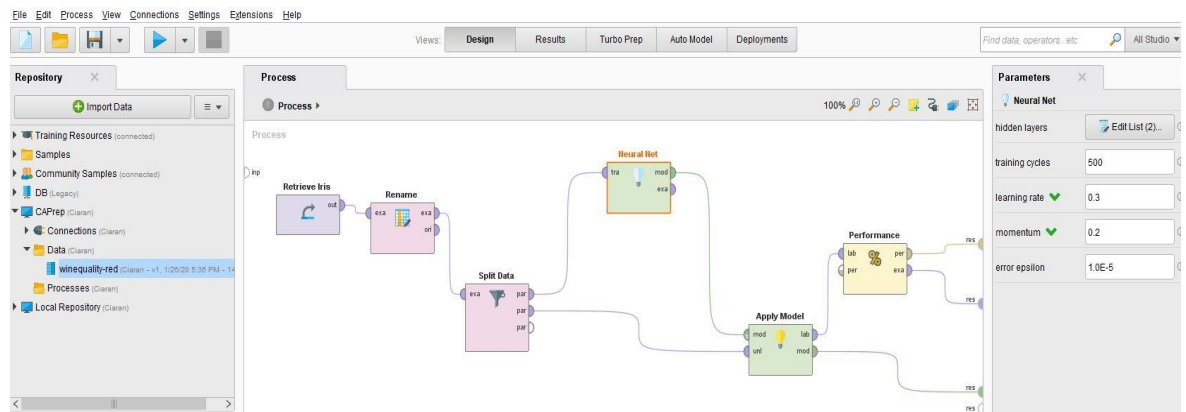


Figure 2

Supplementary data investigation was conducted using a Python project developed in Visual Studio 2019. The structure of the project was modularised around a framework similar to the CRISP-DM model.

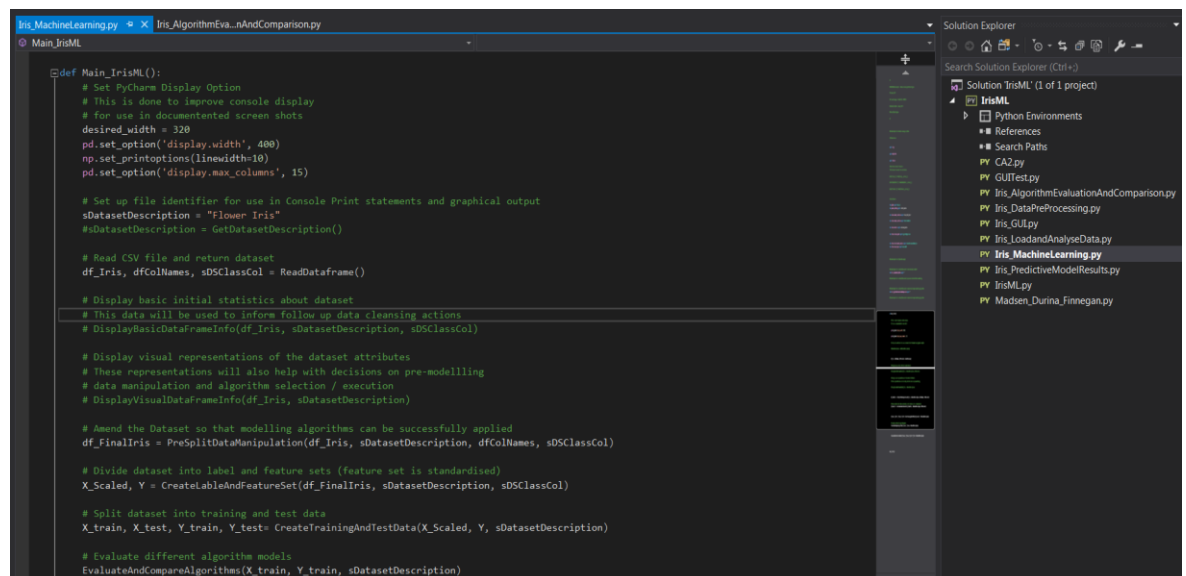


Figure 3

The Python project was used to provide quick-to-develop validations of the RapidMiner process outputs in the Data Understanding phases.



2 Business Understanding

2.1 Determine Business Objectives

The first objective of the data analyst is to thoroughly understand, from a business perspective, what the customer really wants to accomplish.

In our QA, the ‘customer’ and ‘business’ are obviously a theoretical concept. However, given that our chosen dataset relates to Wine Quality, we are assuming the role in the project of a chain of Off-Licence shops, who have a particularly speciality in selling Portuguese “Vinho Verde” red wine. Part of the business USP (unique selling point) is that staff are encouraged to be knowledgeable about the quality of this wine that they may recommend to customers. Although our imaginary Off-Licence chain promotes an awareness of wine amongst staff, very few employees would aspire to the level of sommelier in this brand of Portuguese wine and therefore it is necessary to provide guidance to staff when new stocks of wine arrive in-store.

Vineyards and Wine wholesalers will presumably provide recommendations on what wines are ‘good’ but a secondary objective of our business is to have a less subjective measure of quality for new wines. Our predictive model will therefore provide a more scientific basis for a quality rating, which can be applied across the entire outlet of shops, rather than relying on a human analysis, which could be open to interpretation.

A further secondary objective is that this model may provide guidance for similar in-store marketing of other ‘niche’ wine brands, should our business wish to replicate this approach to wine promotion.

How do we define success? A model is built based to predict the quality of new stocks of red wine based on the constituent chemical properties of the liquid. Our initial dataset will contain information to train and test our model (after various algorithm selections), and we will then conduct an additional test with new ‘unseen’ data to show that the model works well (or not) to provide an employee guide to wine quality.



2.2 Assess Situation

This task involves more detailed fact-finding about all of the resources, constraints, assumptions, and other factors to be considered in determining the data analysis goals and project plan.

Dataset Inventory

We chose a dataset provided in the Kaggle website (<https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009>) that relates to the red wine variant of the Portuguese “Vinho Verde” red wine. (The primary source of the dataset is on the UCI Machine Learning Repository - <https://archive.ics.uci.edu/ml/datasets/wine+quality>).

The dataset is publically available and provides a series of input variables, which are based on physiochemical tests, and an output variable that is a 0 – 10 score of quality.

Assumptions

The ‘quality’ measure, which is obviously the key characteristic we want to assess, has been assumed to come from feedback from wine industry specialists.

Constraints

This an assessment of quality based on the chemical constituents of the wine. There is no data relating to year or grape type, as might be expected with an assessment of wine, so we are assuming that a chemical analysis will provide the all the data points we need for a quantifiable assessment of quality.

There is also no indication of brand or price. This dataset is deliberately excluding these factors (or is unable to include them). Therefore that type of marketing data points will not influence the prediction of ‘quality’ as produced by our model.



2.3 Determine Data Mining Goals

A data mining goal states a project objective in technical terms.

Goal

Our CA project aim is to build a predictive model that provides a categorical rating for a wine based a list of 11 chemical attributes in the liquid.

We extended the modelling process so that a score of '7' or greater is described as 'Very Good', '5 – 6' receives a 'Medium' description, and anything else is 'Poor'. Thus we refine our classification of the model outputs into simpler terms for the end user employees.

Success Criteria

Ideally, we want our model to operate with a greater than 80% accuracy in its predictions of wine quality for new "Vinho Verde" red wines.

2.4 Produce Project Plan

Project Plan

The framework of this document, even just reading from the Table of Contents, provides the general outline of activity.

In brief, our timelines are to complete the following activity by the following milestones (allowing for some iterations and back and forth before project completion);

Any project plan is a dynamic document, and this CA is no exception and we expected, and encountered, the need for many revisions.

- **Saturday January 25th:** Complete dataset selection and establish business objectives.
- **Saturday February 1st:** Complete Data Understanding, Data Preparation, and preliminary model assessment.
- **Saturday February 8th:** Complete Modelling and Evaluation, determine Production approach. Present to class.
- **Sunday February 9th:** Submit CA final report with recommendations.



Assessment of Tools

In order to gain an insight into commonly used industry tools, the majority of the data mining approach was conducted in **RapidMiner** (as can be seen in the screenshots used throughout this document).

However, early stage data analysis and some preparation used **Python** scripting. This was partially because of familiarity with Python from earlier CA work on the course and also to provide some quick additional verification of the RapidMiner outputs.

Excel was used for part of the Data Exploratory Phase, and in the Data Preparation Phase to assist with calculations and understanding of data balancing requirements.



3 Data Understanding

3.1 Collect Initial Data

This involves the acquisition of data and loading into our chosen data mining tool kits.

Initial Data Collection Report

As described in Section 2.2 of this document the dataset for the CA is taken from the Kaggle website, specifically from the URL: <https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009>, which in turn references the original UCI Machine Learning source.



Figure 4

In Appendix A of this document there is a brief guide to understanding wine types and composition, with particular relevance to the chemical data points in this dataset. (Source: <https://github.com/dipanjanS>).

Downloading the CSV file from Kaggle is a straightforward exercise and the CSV file itself is just 101 kB.

For preliminary data analysis the CVS file on Red Wine quality loads without issue into RapidMiner.

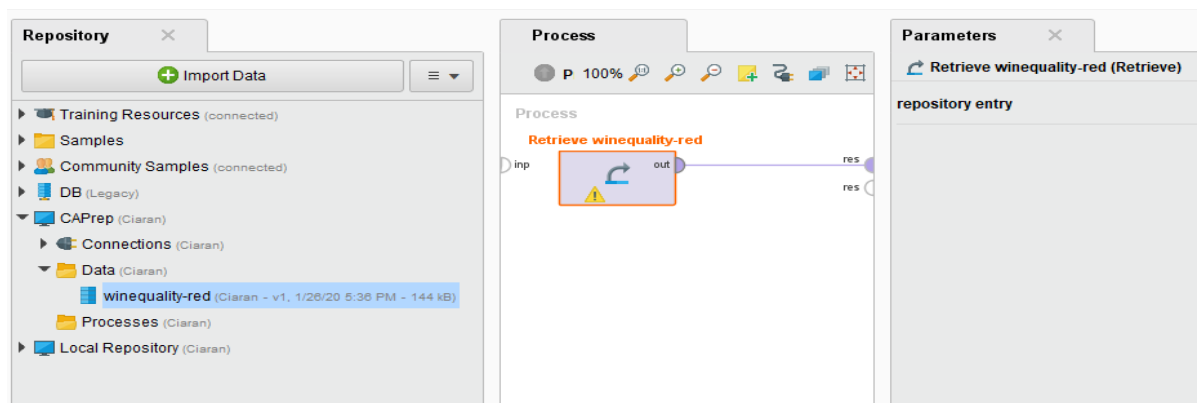


Figure 5



3.2 Describe Data

This involves an examination of the ‘gross’ (or ‘surface’) data and a report on the results.

Data Description Report

The file is on a CSV format, and contains 1600 row with 12 attribute columns in the following structure:

Input variables (based on physicochemical tests):

- 1 - fixed acidity
- 2 - volatile acidity
- 3 - citric acid
- 4 - residual sugar
- 5 - chlorides
- 6 - free sulphur dioxide
- 7 - total sulphur dioxide
- 8 - density
- 9 - pH
- 10 - sulphates
- 11 – alcohol

Output variable (based on sensory data):

- 12 - quality (score between 0 and 10)

A surface view in NotePad++ shows the following sample structure;

```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
winequality-red.csv
1 fixed acidity,volatile acidity,citric acid,residual sugar,chlorides,free sulphur dioxide,total sulphur dioxide,density,pH,sulphates,alcohol,quality
2 7.4,0.7,0.0,1.9,0.076,11.0,34.0,0.9978,3.51,0.56,9.4,5
3 7.8,0.88,0.0,2.6,0.098,25.0,67.0,0.9968,3.2,0.68,9.8,5
4 7.8,0.76,0.04,2.3,0.092,15.0,54.0,0.997,3.26,0.65,9.8,5
5 11.2,0.28,0.56,1.9,0.075,17.0,60.0,0.998,3.16,0.58,9.8,6
6 7.4,0.7,0.0,1.9,0.076,11.0,34.0,0.9978,3.51,0.56,9.4,5
7 7.4,0.66,0.0,1.8,0.075,13.0,40.0,0.9978,3.51,0.56,9.4,5
8 7.9,0.6,0.06,1.6,0.069,15.0,59.0,0.9964,3.3,0.46,9.4,5
9 7.3,0.65,0.0,1.2,0.065,15.0,21.0,0.9946,3.39,0.47,10.0,7
10 7.8,0.58,0.02,2.0,0.073,9.0,18.0,0.9968,3.36,0.57,9.5,7
11 7.5,0.5,0.36,6.1,0.071,17.0,102.0,0.9978,3.35,0.8,10.5,5
12 6.7,0.58,0.08,1.8,0.09699999999999999,15.0,65.0,0.9959,3.28,0.54,9.2,5
13 7.5,0.5,0.36,6.1,0.071,17.0,102.0,0.9978,3.35,0.8,10.5,5
14 5.6,0.615,0.0,1.6,0.089000000000000001,16.0,59.0,0.9943,3.58,0.52,9.9,5
```

Figure 6

In line with the description of the dataset in Kaggle, the dataset contains header information but the remainder of the dataset is purely numeric.

Given that this is a dataset aimed at relative newcomers to the work of Machine Learning, it does not seem likely that there will be any invalid or missing data entries. We would expect this to be borne out in the analysis in the following sections of this document.



3.3 Explore Data

This task addresses data mining questions using querying, visualization, and reporting techniques.

Explore 'Wine Quality' Data

Assessing the Target attribute.

The figure below shows the head rows in the dataset, in a much more readable RapidMiner generated presentation (as compared to Notepad++);

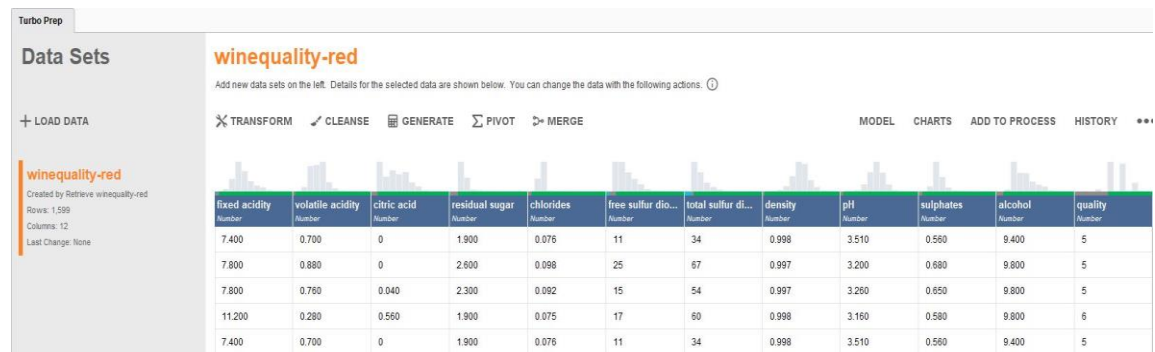


Figure 7

As described earlier, there are 1600 rows in the dataset (one row with header information), and 12 columns.

The purpose of our data mining task is to assess wine quality. Therefore the attribute column for '**quality**' will be marked as our **label**. This is the value which we ultimately want to predict, based on the entries in the other attribute rows.

As will be explained in Section 4 of this CA report, we will carry out feature engineering to add an additional attribute to make the interpretation of the model easier for the end user by moving away from a numeric output to a qualitative description.

Relationships between attributes. Correlations of attributes to the label value and between attributes themselves.

In many datasets not every attribute will have a strong impact on the target label. The values of certain attributes may influence the model very little and can actually be ignored in the model building process.

In addition, certain attributes may be strongly correlated with each other, either negatively or positively. Commonly cited examples in data modelling are if measurements are stored using different standards – for example metric vs imperial. One of those attributes will be redundant because an increase in one attribute is reflected with a similar scale of increase in the other.

For our wine quality dataset we can execute a correlation analysis on the attributes to determine which attributes have the greatest impact on the resultant quality of the wine.



RapidMiner provides a process to carry out this analysis, either through a stand-alone process using the Correlation Metric Operator, or through the AutoModelling process.

After conducting such an analysis on our database, we can see the following correlation data between attributes and label;

Weights by Correlation

Attribute	Weight
alcohol	0.476
volatile acidity	0.391
sulphates	0.251
citric acid	0.226
total sulfur dioxide	0.185
density	0.175
chlorides	0.129
fixed acidity	0.124
pH	0.058
free sulfur dioxide	0.051
residual sugar	0.014

Figure 8

Alcohol content has the greatest impact on the resultant measure of 'quality'.

Residual sugar appears to have the least impact on the determination of red wine quality in our sample dataset.

The diagram above shows the other relative metrics for each attribute in terms of their influence on 'quality'.

Within the attributes themselves, correlations can be examined to determine if some attributes can be considered redundant or, at the very least, less important to include in the model calculations.

The diagram below shows the inter attribute correlations;

Correlations

Attributes	alcohol	chlorides	citric acid	density	fixed acidity	free sulfur dioxide	pH	quality	residual sugar	sulphates	total sulfur dioxide	volatile acidity
alcohol	1	-0.221	0.110	-0.496	-0.062	-0.069	0.206	0.476	0.042	0.094	-0.206	-0.202
chlorides	-0.221	1	0.204	0.201	0.094	0.006	-0.265	-0.129	0.056	0.371	0.047	0.061
citric acid	0.110	0.204	1	0.365	0.672	-0.061	-0.542	0.226	0.144	0.313	0.036	-0.552
density	-0.496	0.201	0.365	1	0.668	-0.022	-0.342	-0.175	0.355	0.149	0.071	0.022
fixed acidity	-0.062	0.094	0.672	0.668	1	-0.154	-0.683	0.124	0.115	0.183	-0.113	-0.256
free sulfur dioxide	-0.069	0.006	-0.061	-0.022	-0.154	1	0.070	-0.051	0.187	0.052	0.668	-0.011
pH	0.206	-0.265	-0.542	-0.342	-0.683	0.070	1	-0.058	-0.086	-0.197	-0.066	0.235
quality	0.476	-0.129	0.226	-0.175	0.124	-0.051	-0.058	1	0.014	0.251	-0.185	-0.391
residual sugar	0.042	0.056	0.144	0.355	0.115	0.187	-0.086	0.014	1	0.006	0.203	0.002
sulphates	0.094	0.371	0.313	0.149	0.183	0.052	-0.197	0.251	0.006	1	0.043	-0.261
total sulfur dioxide	-0.206	0.047	0.036	0.071	-0.113	0.668	-0.066	-0.185	0.203	0.043	1	0.076
volatile acidity	-0.202	0.061	-0.552	0.022	-0.256	-0.011	0.235	-0.391	0.002	-0.261	0.076	1

Figure 9



The darker colours indicate attributes in the feature set that have higher correlation values.

Not surprisingly, the **pH** and **fixed acidity** attributes are showing more marked negative correlation than many of the other features.

Likewise the **free sulphur dioxide** and **total sulphur dioxide** features are showing a reasonably positive correlation.

Although the Wine Quality dataset is not large in terms of rows and attributes, the principle holds that models can be improved in terms of creation and execution if the feature set is pruned to just those attributes that have the greatest impact in the model accuracy.

Taking the above considerations into account the list of attributes with which to train our potential models, which is described in detail in Section 5 of this document, can be pruned to improve performance.

Simple Statistical Analysis of the Wine Quality Dataset

RapidMiner provides a simple and graphical means to generate some basic statistical data on the attributes in the dataset.

The figures below provide a visual description of the histograms for each attribute in the Wine Quality dataset.

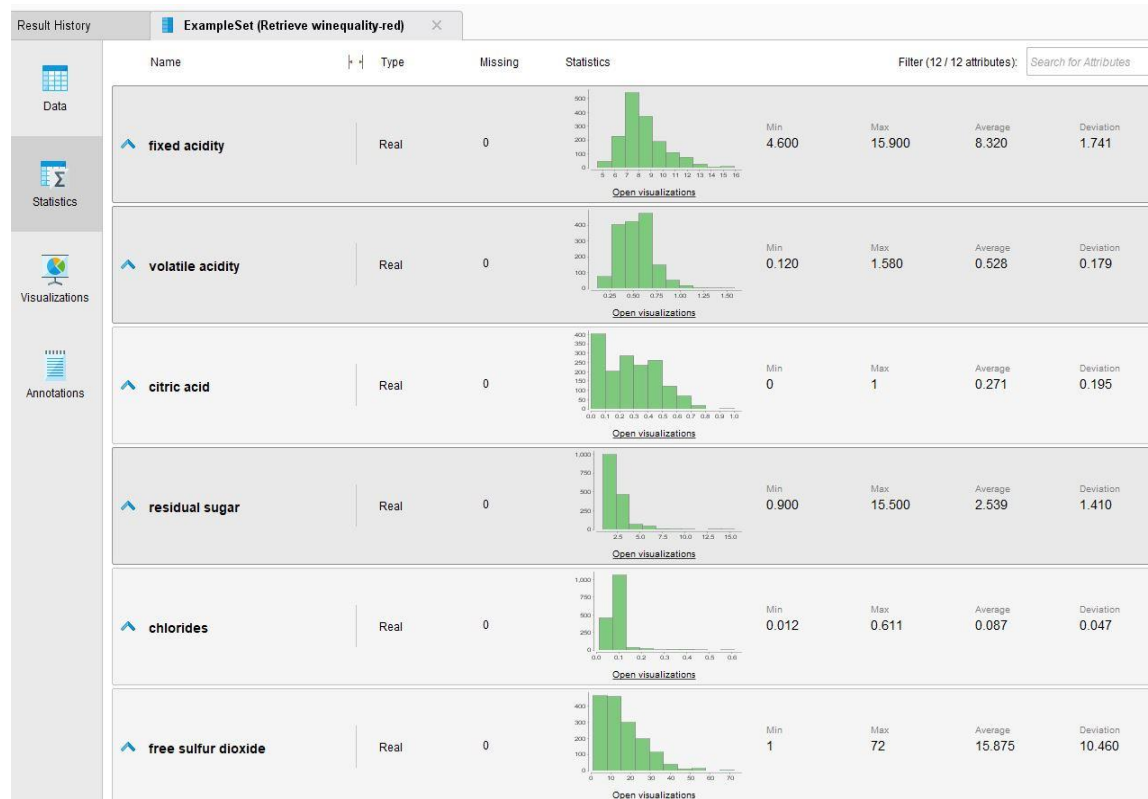


Figure 10

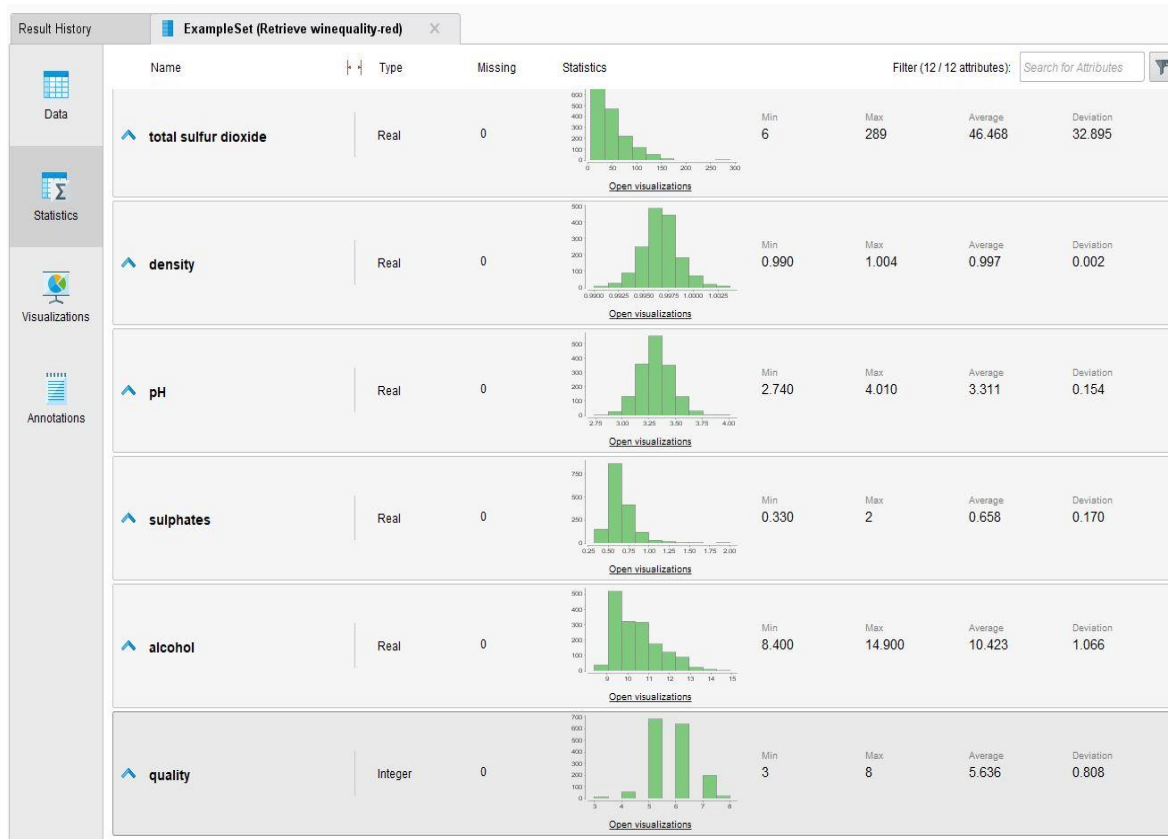


Figure 11



Data Exploration Report

Initial findings / hypotheses and their impact on the remainder of the Wine Quality project.

Standardise and normalise the feature attributes

All the values in the dataset are numeric. This will simplify some of the data preparation tasks, as the data mining process needs numeric data to build a mathematical model.

However, although the range of numeric data is not particularly large there are still some features that use a noticeably difference scale, for example the free sulphur dioxide range of values.

In order to prevent such attributes from skewing the resultant data model, all the elements in the feature set will be normalized. This process is described in more detail in Section 4 of this report.

Quality of Data – No Missing Rows

The following three screen shots are from the AutoModel output of Rapid Miner. They are a sample of the 'General' output from the AutoModel process.

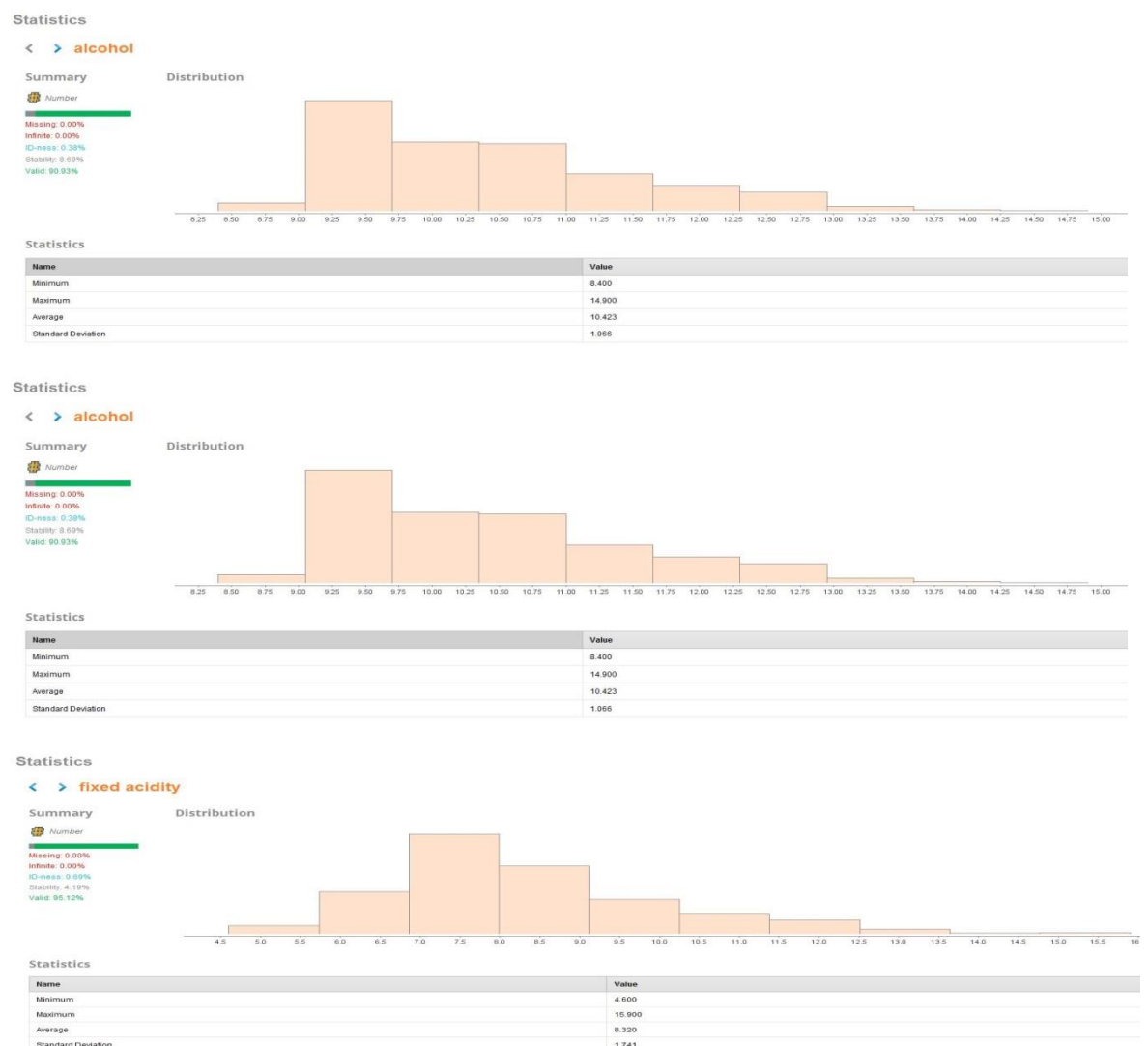


Figure 12



The RapidMiner analysis shows that there are no missing data elements in any of the attributes in the dataset.

This will reduce the complexity in the data preparation stage of the project as it will not be necessary to impute missing data, nor will it be necessary to remove incomplete rows from the dataset.

We can see this again in the list of attributes in the 'Result's tab for the dataset in RapidMiner.

Name	Type	Missing	Min	Max	Average
fixed acidity	Real	0	4.600	15.900	8.320
volatile acidity	Real	0	0.120	1.580	0.528
citric acid	Real	0	0	1	0.271
residual sugar	Real	0	0.900	15.500	2.539
chlorides	Real	0	0.012	0.611	0.087
free sulfur dioxide	Real	0	1	72	15.875
total sulfur dioxide	Real	0	6	289	46.468
density	Real	0	0.990	1.004	0.997
pH	Real	0	2.740	4.010	3.311
sulphates	Real	0	0.330	2	0.658
alcohol	Real	0	8.400	14.900	10.423
quality	Integer	0	3	8	5.636

Showing attributes 1 - 12

Figure 13

Quality of Data – Duplicate Values

A quick Python based routine was run independently to confirm that no rows in the Wine Quality dataset are duplicates.



Quality of Data – Zero Values

However, although our analysis shows no missing rows or obvious errors in the format of data in the columns, we ran an additional check for 'zero' values.

A zero entry, particularly when there is a significant proportion of information in a dataset that is in numeric format, can also be an indication of missing data.

A quick EXCEL analysis of the Wine Quality csv file will show that there are **132** rows for the 'citric acid' attribute that have a zero value.

A quick validation in our supplementary Python program will show a similar analysis.

Is this a legitimate data entry, or are these data rows incomplete?

To investigate further we referenced supporting material on Kaggle, and elsewhere, that explains the fermentation process for wine in more detail. **Citric acid** is typically only found in small quantities in wine, and it is used to add what is described as 'freshness' to the wine by enhancing flavour.

It is not unusual that the citric acid component in a given type of wine to be completely consumed during the wine fermentation process, and it is not always added back into the process.

Therefore we have concluded that a 'zero' entry for citric acid is a valid data point and will not need to be addressed in the data preparation phase of this project.



Balancing the Wine Quality Dataset

There is a concern on the spread of wines in the dataset in terms of quality.

RapidMiner allows for a more detailed bar chart view on the 'quality' data in the Wine Quality dataset.

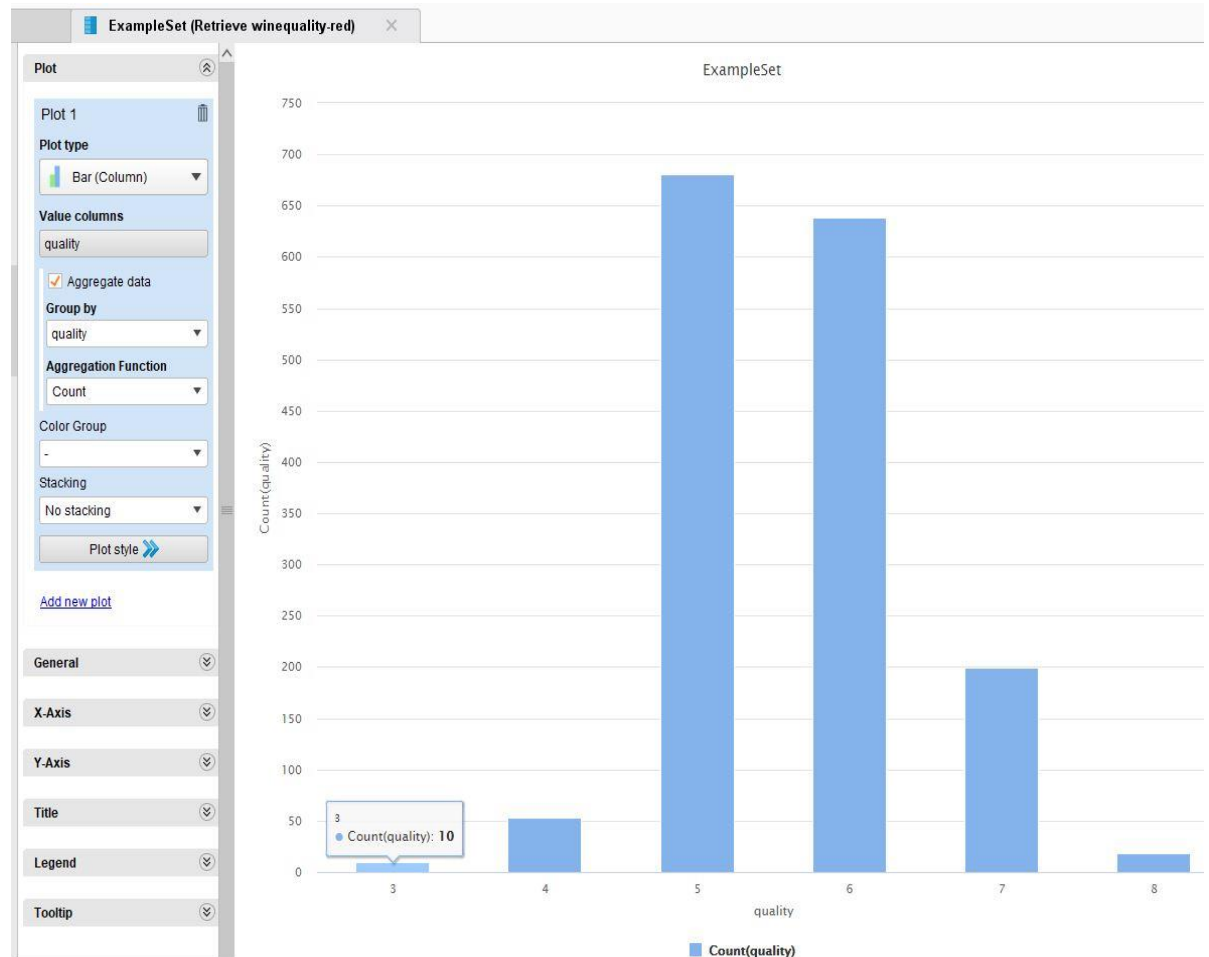


Figure 14

There is an immediate concern that the dataset has a high volume count of wines with a 'quality' rating of 5 or 6. However there are relatively few of either the very low or very high quality red wine types.

The Wine Quality dataset is unbalanced and does not have an even spread of wine qualities. This will impact on the accuracy of any model we attempt to build.

Section 4 of this report looks at steps we need to take to balance the Wine Quality dataset before we attempt to train and evaluate any models.



The bar chart above provides a good graphical representation of the spread.

For a simpler 'at-a-glance' view the Python project output at this phase of data exploration shows the following distribution of 'quality' classes.

```
# Class distribution - present the count of the number of rows that
# belong to each class in the dataset
print("\n\t{} Dataset Class Distribution : \n".format(datasetDescription))
print(dataset.groupby(sClass).size())
```

Output

Show output from: Debug

Vinho Verde Red Wine Quality Dataset Class Distribution :

quality	
3	10
4	53
5	681
6	638
7	199
8	18
quality	1
dtype:	int64

Figure 15



3.4 Verify Data Quality

Is the data complete? Does it contain errors and/or missing data? If so, how common are these issues?

Data Quality Report

Data quality has been shown to be very good based on the source description and our own data analysis.

There are no missing entries and no obvious errors in the dataset. There are no documented errors in the comments section on the Kaggle page from which the dataset was downloaded and this provides additional assurances.

Checking for Outliers

It is impractical to visually assess outliers with the numbers of features in the Wine Quality dataset.

However, RapidMiner provides Operators to detect outliers in datasets.

The following simple RapidMiner process was created to use Local Outlier Factors to generate outlier scores for each row in the Wine Quality dataset.

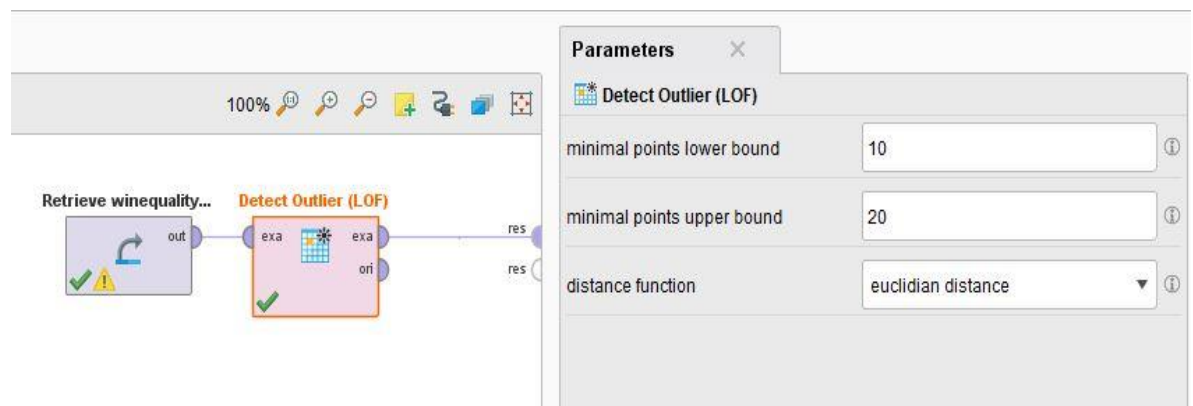


Figure 16

The data shows two distinct outliers in the result set (at the top of the result set below).

Result History														ExampleSet (Detect Outlier (LOF))	
														Open in	
														Turbo Prep	
														Auto Model	
														Filter (1,599 / 1,599 examples):	
														all	
Row No.	quality	outlier ↓	fixed acidity	volatile acidity	citric acid	residual sug...	chlorides	free sulfur d...	total sulfur d...	density	pH	sulphates	alcohol		
1082	7	12.379	7.900	0.300	0.680	8.300	0.050	37.500	289	0.993	3.010	0.510	12.300		
1080	7	11.399	7.900	0.300	0.680	8.300	0.050	37.500	278	0.993	3.010	0.510	12.300		
481	5	3.847	10.600	0.280	0.390	15.500	0.069	6	23	1.003	3.120	0.660	9.200		
1245	6	3.140	5.900	0.290	0.250	13.400	0.067	72	160	0.997	3.330	0.540	10.300		
1039	7	2.993	8.700	0.410	0.410	6.200	0.078	25	42	0.995	3.240	0.770	12.600		
1044	7	2.885	9.500	0.390	0.410	8.900	0.069	18	39	0.999	3.290	0.810	10.900		
1236	4	2.780	6	0.330	0.320	12.900	0.054	6	113	0.996	3.300	0.560	11.500		
653	5	2.554	15.900	0.360	0.650	7.500	0.096	22	71	0.998	2.980	0.840	14.900		
774	6	2.488	7.900	0.400	0.290	1.800	0.157	1	44	0.997	3.300	0.920	9.500		
918	6	2.442	6.800	0.410	0.310	8.800	0.084	26	45	0.998	3.280	0.640	10.100		

Figure 17



A more graphical representation in RapidMiner displays the outliers in the following Scatter Plot;



Figure 18

A key point to consider is that the 'outliers' are marked as 'high quality' wines, which can be seen by the colour coding of the entries in the scatter plot above.

As already described, there are very few high quality wines in the dataset hence it is logical that this data might appear as outliers.

Deleting the outliers in this dataset would actually remove key information used in the building of the predictive model. As discussed in Section 4 of this document, it will be necessary to augment the dataset by artificially adding 'high quality' data rows to accompany these outliers.



4 Data Preparation

The output of this phase of the project is the creation of an adapted dataset, which will be used for modelling and major analysis.

Our Wine Quality dataset will be transformed into a format that allows effective modelling and evaluation.

4.1 Select Data

Our 'business' is using this Kaggle Wine Quality dataset to answer the requirement to predict the quality of new red wines delivered to our outlet, based on its chemical composition.

Data Volumes and Technical Constraints

The Wine Quality dataset is relatively small, with 1600 rows, so there is no requirement to reduce the number of rows upon which we will build our model in RapidMiner. A dataset this size is not expected to be excessively computationally expensive, even with the more elaborate algorithms we are expected to evaluate.

Our personal laptops should have no practical processing limitation with a dataset of this size and it is therefore not necessary to 'slice' or partition the data in any way.

Quality

There are no quality issues that warrant the removal of any data rows.

Selection of Attributes

There are twelve columns in the original Wine Quality dataset from Kaggle. As such, feature selection might not appear to be a crucial element in data preparation.

However, many datasets contain attributes that have very limited impact on the final values predicted by the type model for which we are searching. Datasets will also often contain attributes that provide close to identical information or are heavily related to each other, and hence introduce a certain redundancy.

It is good practice to eliminate those features that provide relatively limited value, and this is a principle we applied to our Wine Quality dataset.

AutoModel analysis in RapidMiner provided the guideline to a reduced feature set. Section 3.3 of this document provided screen shots of the correlation values attached to each attribute, both in terms of their influence on the 'target' value (quality) and their relationship to each other.



There was a considerable amount of iteration from model selection and evaluation back to feature selection to determine the impact of various attribute lists.

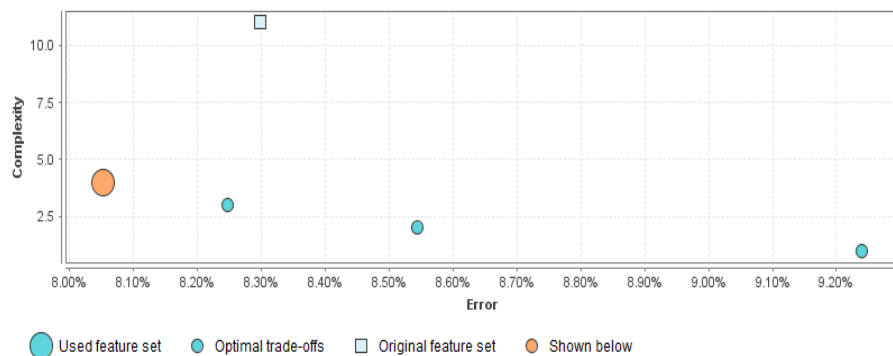
As an example, we looked at the recommended feature set for a 'RandomForest' algorithm used on the Wine Quality dataset and the following list was suggested by Rapid Miner AutoModel;



Random Forest - Feature Sets

Number of Evaluated Feature Sets: 1,235

Optimal Trade-offs between Complexity and Error



Description

The plot on the left shows the result columns or even including some new and achieves an error rate of 18% o validated on a hold-out set will be hi

The best feature sets are in the bott the complexity without increasing the and still more accurate than the origi set which has been used to build the

You can click on each dot to see the importance of features for the model

Currently Selected Feature Set

Name	Expression
volatile acidity	[volatile acidity]
sulphates	[sulphates]
total sulfur dioxide	[total sulfur dioxide]
alcohol	[alcohol]

Figure 19

A reduced feature set can remove computational complexity when an algorithm is being applied to build a model. A more relevant set of features can also help with the final accuracy of the model. and possibly make it easier to understand some of the underlying workings.

For our Wine Quality dataset the following Feature Selection operator in RapidMiner was put in place within the overall Data Preparation Phase processes.

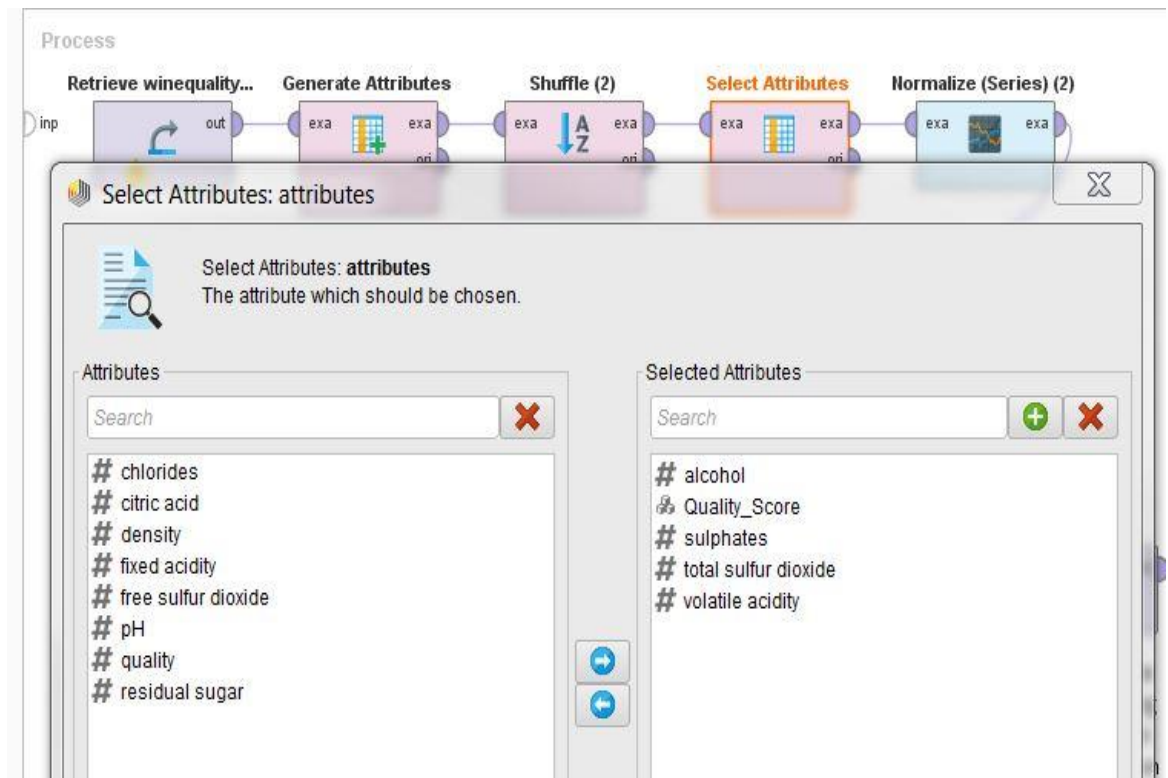


Figure 20

The screenshot above shows an attribute names 'Quality_Score', which is not listed in the description of the original Wine Quality dataset. This is a constructed feature and is described in Section 4.3 of this report.



4.2 Clean Data

In general, this task in the Data Preparation Phase is intended to raise the data quality to the level required by the selected analysis technique.

Clean Data Set

As discussed, data quality and ‘cleanliness’ was not an issue with this Kaggle dataset on Wine Quality.

The numerical data in each attribute was in a consistent format and data type.

There were no categorical attributes in the original dataset that could have introduced error or ambiguity into our modelling process.

The data quality within the Wine Quality dataset allowed us to proceed quickly to the data construction task (Section 4.3) within this Data Preparation Phase.



4.3 Construct Data

This task usually includes constructive data preparation operations such as the production of new derived attributes, or entire new records.

At this stage in the process we may also find it necessary to transform values for existing attributes.

In the sub-sections below, we describe the various tasks we apply to our Wine Quality dataset as part of the re-construction of the data prior to the commencement of the Modelling Phase.

Feature Generation

A business objective of this project was to simplify the output of the predictive model so that employees could quickly assess the quality rating of a new wine in store.

We felt it would be sensible to supplement the 1 – 10 'quality' score with the following categorical descriptions;

- Poor Quality
- Medium Quality
- High Quality

Thus we added an operator in the RapidMiner Data Preparation Phase processes to generate a new attribute called 'Quality_Score'. Our intention was to allow for more meaningful categorization of the data.

The RapidMiner Operator uses the following logic in an attribute generation process;

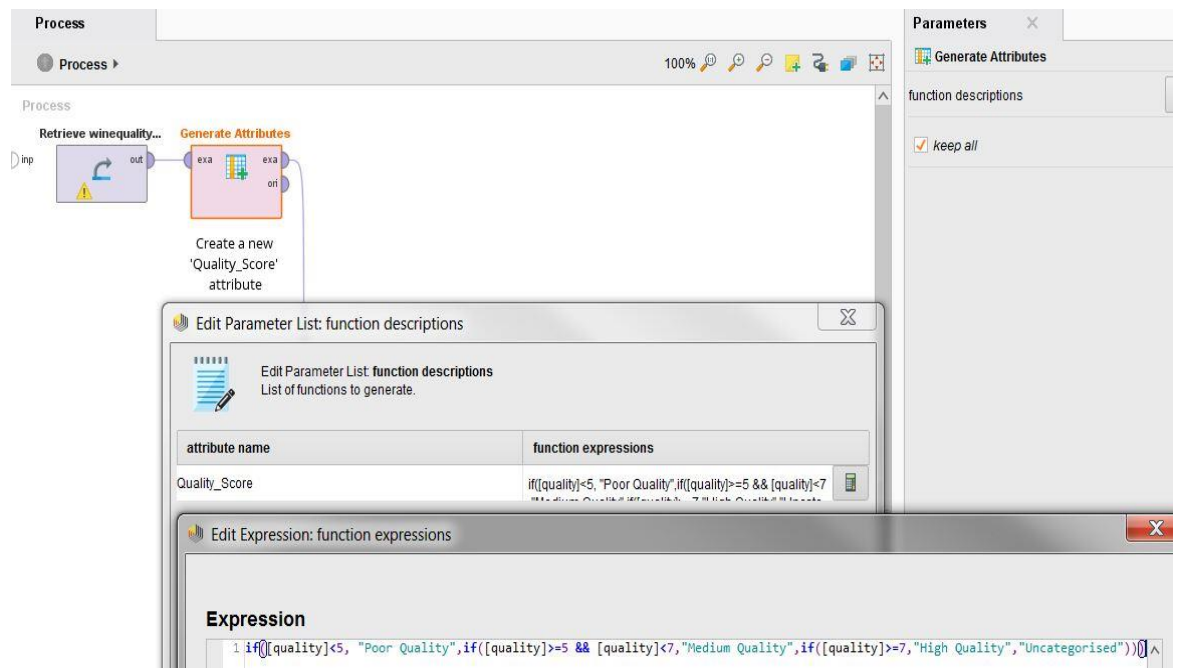


Figure 21

Aside from making the deployed model more straightforward to use in production, we have changed our modelling challenge into a Classification problem. We felt that this would generate a more useful and readable set of analysis on each algorithm in our chosen models.

Linear Regression would generate a predicted real number in the range 1 – 10 but the output on accuracy would not be as readily understandable. In addition, our business objective is to quickly provide a meaningful rating of new wine so it made sense to focus our modelling effort on a Classification challenge.

The new 'Quality_Score' attribute is derived from the 'Quality' value and appended as a new attribute to the Wine Quality dataset by the RapidMiner operator.

Normalisation

Additional good practice in data mining is to normalise the attribute values to prevent bias with larger numerical values.

In our Wine Quality dataset the range of values for *free sulphur dioxide* is 1 to 72 units, and for *total sulphur dioxide* is 6 to 289 units. This is a larger set of absolute numerical values than the other attributes in the dataset.

To avoid values in those features introducing bias we employ a Normalize operator in our RapidMiner Data Preparation Phase process.

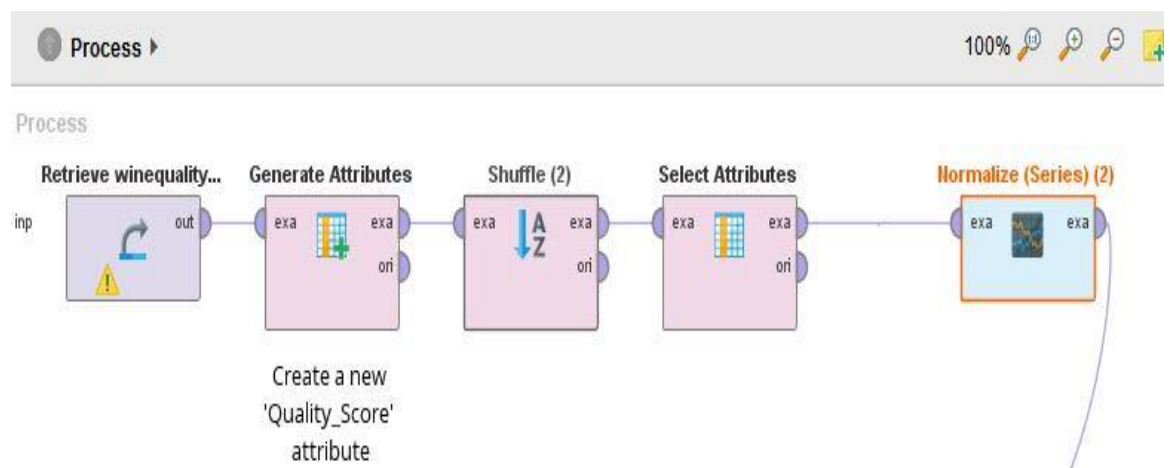


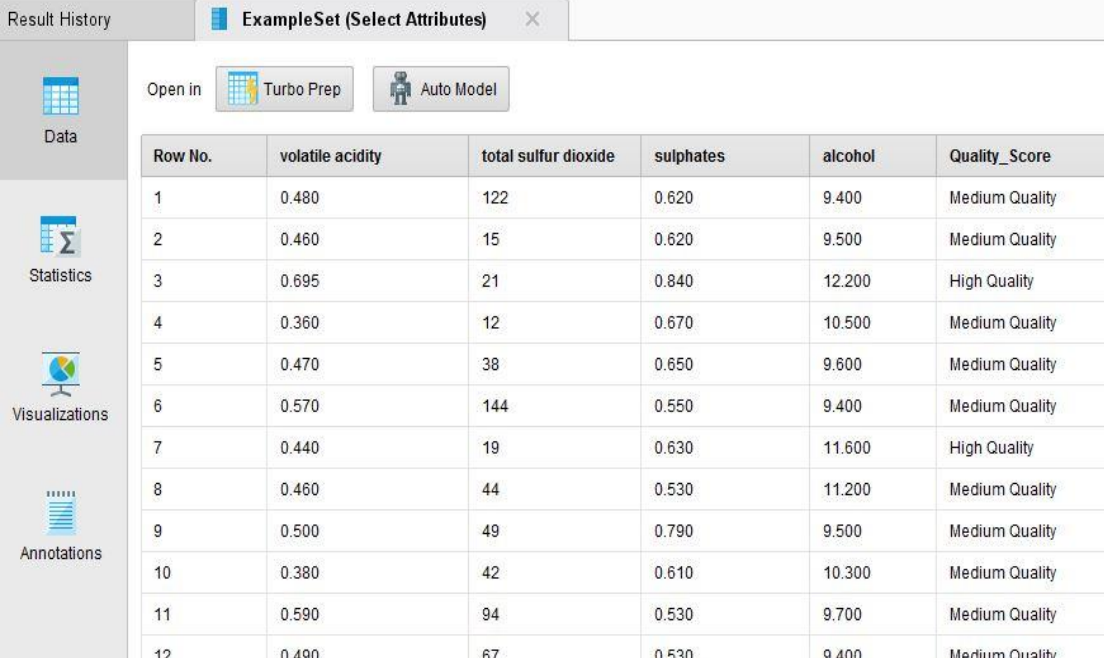
Figure 22

This task 'smooths' out the numerical values in the dataset but preserves the relative differences between the features. It is a purely syntactic change to satisfy the requirements for many modelling algorithms to deliver as accurate a result as possible.



A before and after view of a section of the Wine Quality dataset, as represented in the images below, will display the nature of the data transformation.

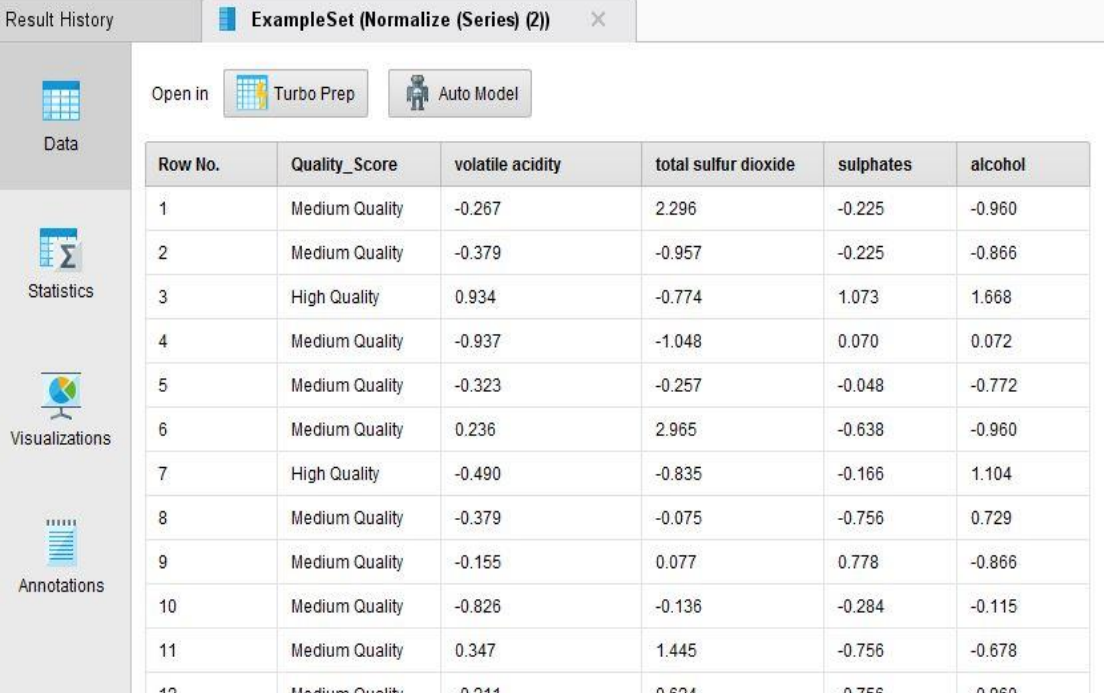
This is a partial view of the Wine Quality dataset before we execute the 'Normalize' operator in RapidMiner (and after the feature set of attributes have been reduced);



Row No.	volatile acidity	total sulfur dioxide	sulphates	alcohol	Quality_Score
1	0.480	122	0.620	9.400	Medium Quality
2	0.460	15	0.620	9.500	Medium Quality
3	0.695	21	0.840	12.200	High Quality
4	0.360	12	0.670	10.500	Medium Quality
5	0.470	38	0.650	9.600	Medium Quality
6	0.570	144	0.550	9.400	Medium Quality
7	0.440	19	0.630	11.600	High Quality
8	0.460	44	0.530	11.200	Medium Quality
9	0.500	49	0.790	9.500	Medium Quality
10	0.380	42	0.610	10.300	Medium Quality
11	0.590	94	0.530	9.700	Medium Quality
12	0.490	67	0.530	9.400	Medium Quality

Figure 23

This is the Wine Quality dataset after we execute the 'Normalize' operator;



Row No.	Quality_Score	volatile acidity	total sulfur dioxide	sulphates	alcohol
1	Medium Quality	-0.267	2.296	-0.225	-0.960
2	Medium Quality	-0.379	-0.957	-0.225	-0.866
3	High Quality	0.934	-0.774	1.073	1.668
4	Medium Quality	-0.937	-1.048	0.070	0.072
5	Medium Quality	-0.323	-0.257	-0.048	-0.772
6	Medium Quality	0.236	2.965	-0.638	-0.960
7	High Quality	-0.490	-0.835	-0.166	1.104
8	Medium Quality	-0.379	-0.075	-0.756	0.729
9	Medium Quality	-0.155	0.077	0.778	-0.866
10	Medium Quality	-0.826	-0.136	-0.284	-0.115
11	Medium Quality	0.347	1.445	-0.756	-0.678
12	Medium Quality	-0.211	0.624	-0.756	-0.960

Figure 24



Data Balancing

This was one of the most significant tasks within the Data Preparation Phase, and across this entire data mining project.

It was necessary to iterate backwards and forwards through the phases on Data Preparation, Modelling, and Evaluation in order to find a balance to the Wine Quality dataset that produced the best results. (In practice, we ended up settling on the 'least bad' approach to data balancing).

This sub-section largely describes the end-point at which we arrived in terms of balancing the Wine Quality dataset, but it illustrates the type of operations we executed on the data for this task.

The final observation in Section 3.3 of the document ('Explore Data') related to the imbalance of wine types in the dataset. There are considerably more '5' and '6' quality wines than wine at either end of the quality spectrum.

Section 4.3 describes how we added an additional attribute to create a classification attribute 'Quality_Score' which groups the 1 – 10 values under three different quality descriptions ('Poor', 'Medium', 'High'). This simplifies the understanding of the classification of the data but does not help with the balance of the data. There are still 5 to 6 times more wines described as 'Medium' than either grouping for 'Poor' or 'High'.

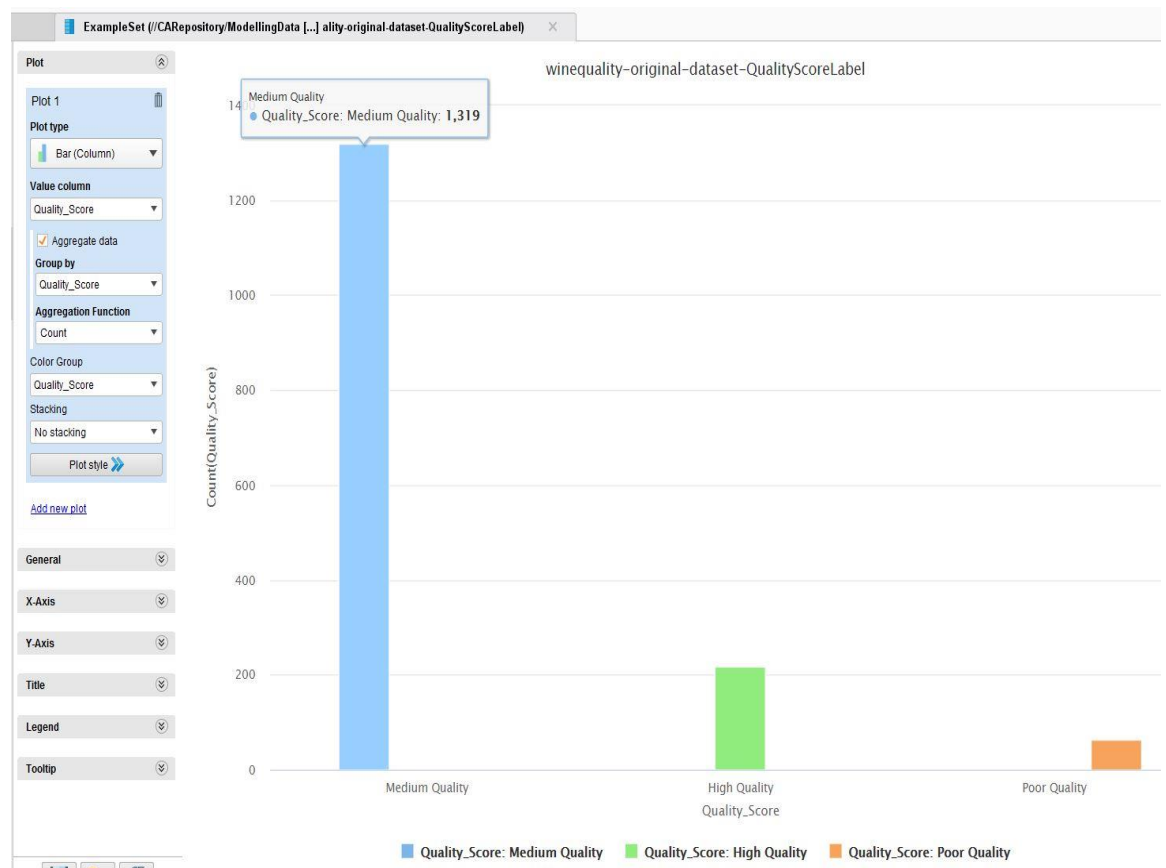


Figure 25



Working with this Wine Quality dataset is likely to create a predictive model that is biased towards classifying wine as 'Medium'. Machine learning algorithms have trouble learning when one class dominates the other, or others as in the case of our Wine Quality dataset. There is not enough data to properly model the characteristics of a new 'Poor' or 'High' quality wine.

What were the potential solutions for the project? The options we considered were;

- Create new 'synthetic' row for the 'Poor' and 'High' quality wines.
- Reduced the number of 'Medium' data rows to be introduced into the Modelling Phase.
- A combination of both the above options.

Upsampling

Our prior data modelling experience suggested the use of a 'SMOTE' (**S**ynthetic **M**inority **O**ver-sampling **T**echnique) approach to generate new rows in the Wine Quality dataset.

RapidMiner, through an additional extension, provides an operator which can be applied to a dataset to create new 'artificial' rows of data, which attempt to mirror the required classification of data.

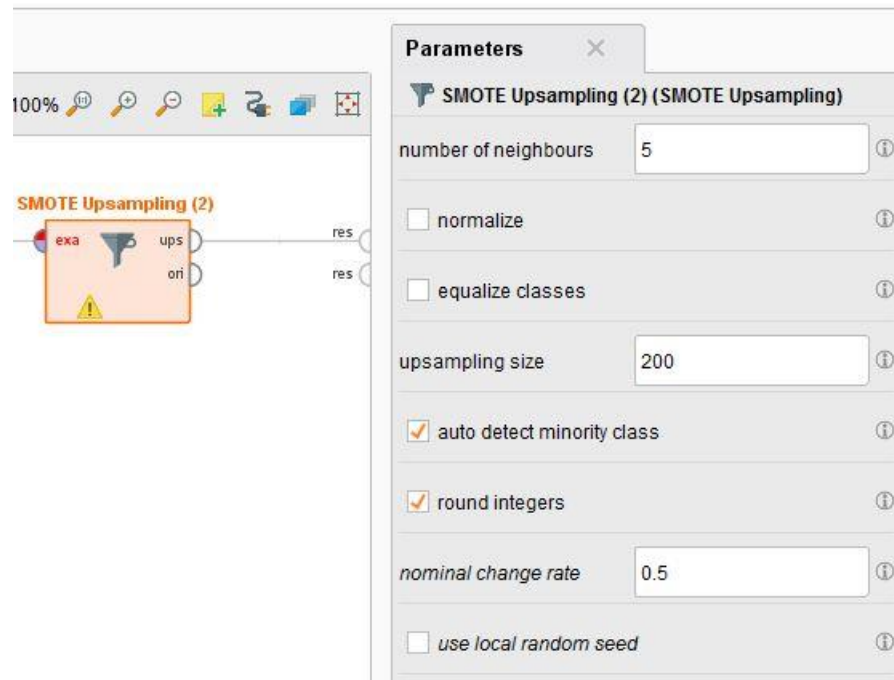


Figure 26



Downsampling

Balancing the Wine Quality dataset can also involve reducing the number of row for 'Medium' quality wines.

A nested set of operations can be embedded into a RapidMiner process to rebalance the data and down sample the 'Medium' wines.

We created a high level 'Unbalance' process operator in RapidMiner into which to feed our Wine Quality dataset.



Figure 27

Clicking through to the workings of the operator we can see how the Wine Quality dataset is split to separate out the 'Medium' Quality wines and reduce the % of those rows fed into the training process for the model.

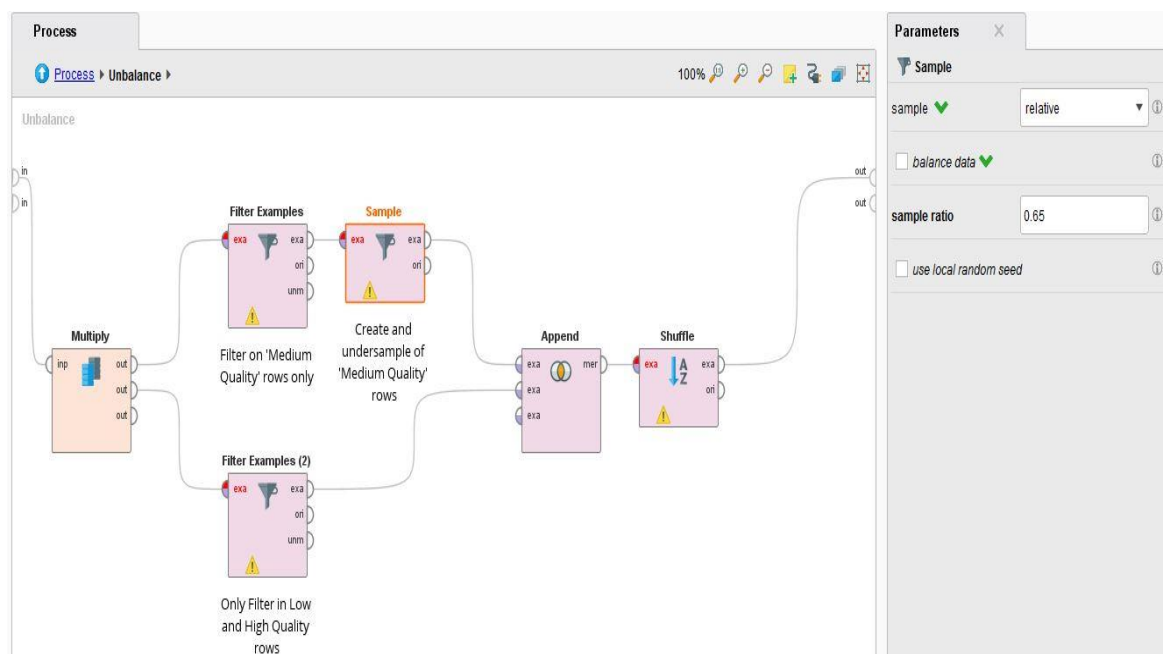


Figure 28

The Upsampling and Downsampling operators are chained within a separate process so that can be more easily re-used with the RapidMiner modelling processes.

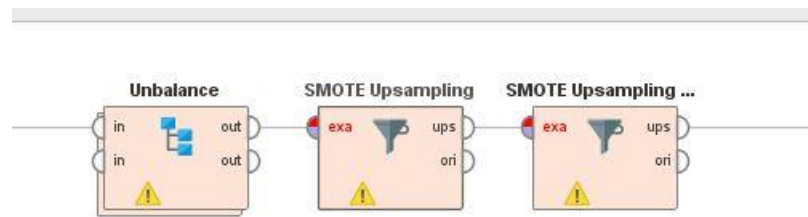


Figure 29

Why two SMOTE operators?

Each SMOTE operator in RapidMiner tackles the minority class, which is the classification with the fewest number of rows.

In our Wine Quality dataset the first SMOTE operator tackles the 'Poor' quality subset of data and generates new rows for that classification.

The second SMOTE operator follows in sequence. As the minority class is now 'High' quality wines, the operator generates new artificial data for that classification.

Using the SMOTE operators in sequence allows for a balancing process to take place on both of the 'minor' classifications.

How much artificial data to create?

The best data to use is information collected in the real world. However, our Wine Quality dataset is deficient in this area, with 'Medium' quality data rows very clearly dominant.

The default setting in the SMOTE operator in RapidMiner is to generate enough data to balance the minority class with the majority one.

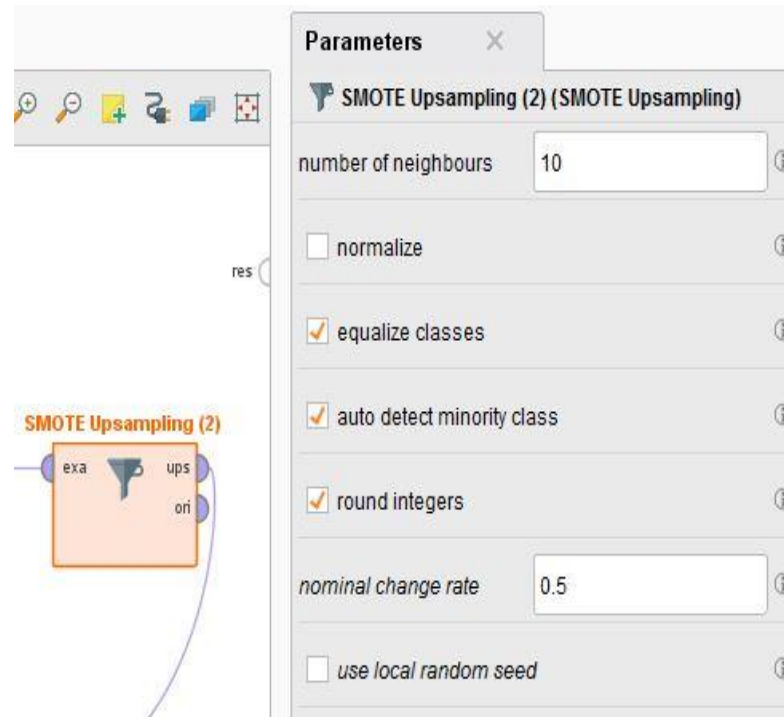


Figure 30

Applying this sequence in parallel would generate many times more artificial data in the Training Set than 'real data', and would not be desirable.

This setting was tested with our Wine Quality dataset but the resultant models performed poorly on Test data.

We looked at the following numerical balance in the data and determined what we felt would be an acceptable level of new 'synthetic' data.

The Wine Quality dataset was split into Training and Test data prior to modelling with a 70%/30% ratio. The numbers of rows in each set breakdown as follows;

Overall	1599	Split
Training	1119	0.7
Test	480	0.3



The distribution of rows, based on quality, within the Training and Test data is;

<i>Wine Quality</i>	Training (No SMOTE)	Test
Poor Quality	41	22
Medium Quality	924	395
High Quality	154	63
	1119	480

The SMOTE Operator allows the user to select an upper maximum number of new artificial rows.

We experimented with the following options;

<i>Wine Quality</i>	UpSample Opt - 1	UpSample Opt - 2	UpSample Opt - 3
Poor Quality	300	200	100
Medium Quality			
High Quality	300	200	100
	<i>Percentage Upsampling in Training Set</i>		
	54%	36%	18%

The above numbers – 100, 200, and 300 – represent different settings in the SMOTE operator configuration. They represent an upper limit to the additional number of artificial data rows.

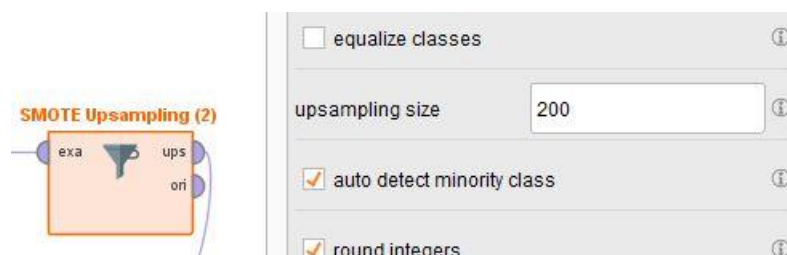


Figure 31



The table above shows the effect that the configuration values have on the actual percentage of artificial records in the Training Set, before it is used in the Modelling Phase.

The result in terms of actual new 'synthetic' rows is given in the table below.

	Training (with SMOTE) Opt 1	Training (with SMOTE) Opt 2	Training (with SMOTE) Opt 3
<i>Wine Quality</i>			
	<i>+300 minority rows</i>	<i>+200 minority rows</i>	<i>+200 minority rows</i>
Poor Quality	341	241	241
Medium Quality	924	924	924
High Quality	454	354	154
Total	1719	1519	1319

Our initial assumption was that 50%+ of artificial data in the Training Set was too high a value, and that 18% of an increase still generated too few 'Poor' and 'High' quality data rows for the Modelling.

We choose to set the SMOTE Operator parameter in RapidMiner at '200', generating just over 35% of new records in advance of Modelling.

During the actual project we iterated a number of times with the value in the SMOTE operators and the '200' value performed the best on the Test set. (However, as stated earlier in this report this was really just the 'least worst' setting in terms of generating predictive values for 'Poor' and 'High' quality wines).

How much to remove?

The scale of Downsampling was much more of a trial and error process as we iterated through the Data Preparation, Modelling, and Evaluation phases to tune the models to try and maximise accuracy.

In practice we found that Downsampling the 'Medium' wines in the Training set to **65%** of the original number produced marginally better accuracy in the Modelling / Evaluation phase.

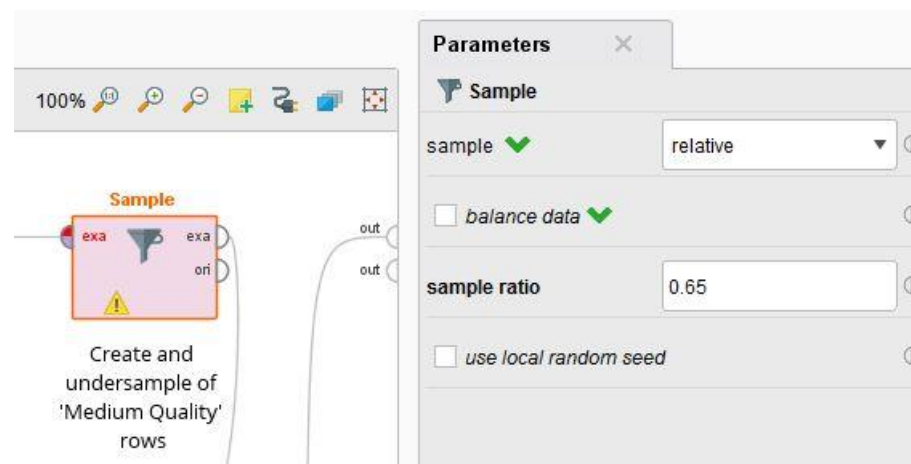


Figure 32

The Upsampling and Downsampling routines are built and refined as part of the Data Preparation Phase but implemented during the training of the Model.

It is important to note that the SMOTE and down sampling operations are **only** applied to the Training dataset for Wine Quality. This is the subset of the overall Wine Quality dataset used to train the actual model.

The application of these over and undersampling routines within the Modelling Phase is described in detail in Section 0 and 5.3 of this document.



4.4 Integrate Data

Our Wine Quality dataset from Kaggle is a complete repository of information for our data mining purposes.

We determined that there is no need to merge additional data sources, although we do carry out a supplementary 'White Wine' analysis after the main modelling, evaluation and deployment.

See Appendix B for details.

4.5 Format Data

Formatting transformations in this task primarily refer to syntactic changes made to the data that do not change its meaning, but may be required by the modelling tool and/or choice of modelling algorithm.

The major tasks in reworking the Wine Quality dataset are described in Section 4.3, but there are also some other minor updates that we make to the data

Data Ordering

The Kaggle page describes the Wine Quality dataset as 'ordered'.

In data mining it is often important to change the order of the records in the dataset. Many modelling algorithms will need datasets to be in a fairly random order. For example, when using neural networks it is generally best for the records to be presented in a random order.

We do not employ the use of neural networks in this project but, following good practice, we deployed a Shuffle operator in our RapidMiner Data Preparation Phase processes.

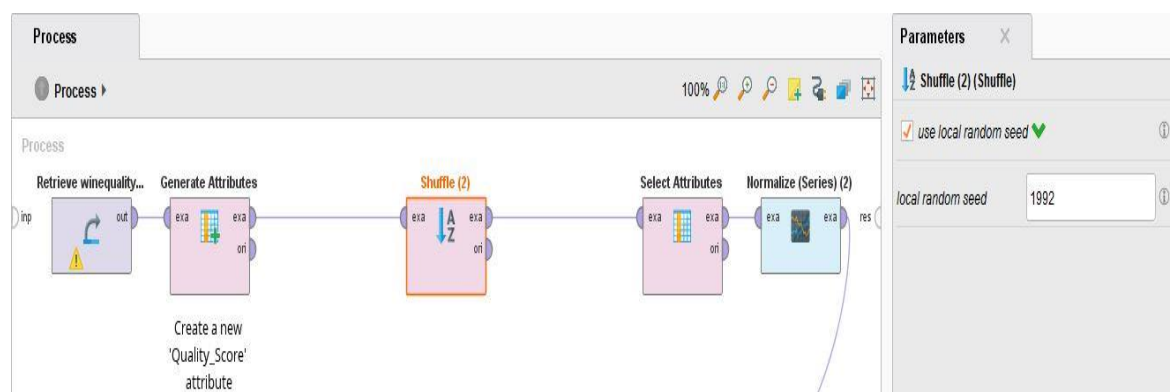


Figure 33

By shuffling the data, the intention is to remove potential bias that could be introduced by the sequence with which the rows in our Wine Quality dataset were added.



5 Modelling

5.1 Select Modelling Technique

The first step in the Modelling Phase is to select the actual modelling technique to be used.

This is a Classification problem so there are many academic guidelines. We also choose to focus on using RapidMiner exclusively as our tool of choice from this point on in the project and this influenced our approach to the Modelling Phase.

The RapidMiner website provides resources (<https://mod.rapidminer.com/>) to suggest modelling techniques based on the characteristics of a dataset. Based on the nature of the Wine Quality dataset this web based RapidMiner application suggested the following models (this is a partial screenshot with the highest rated models on the right in **red**):

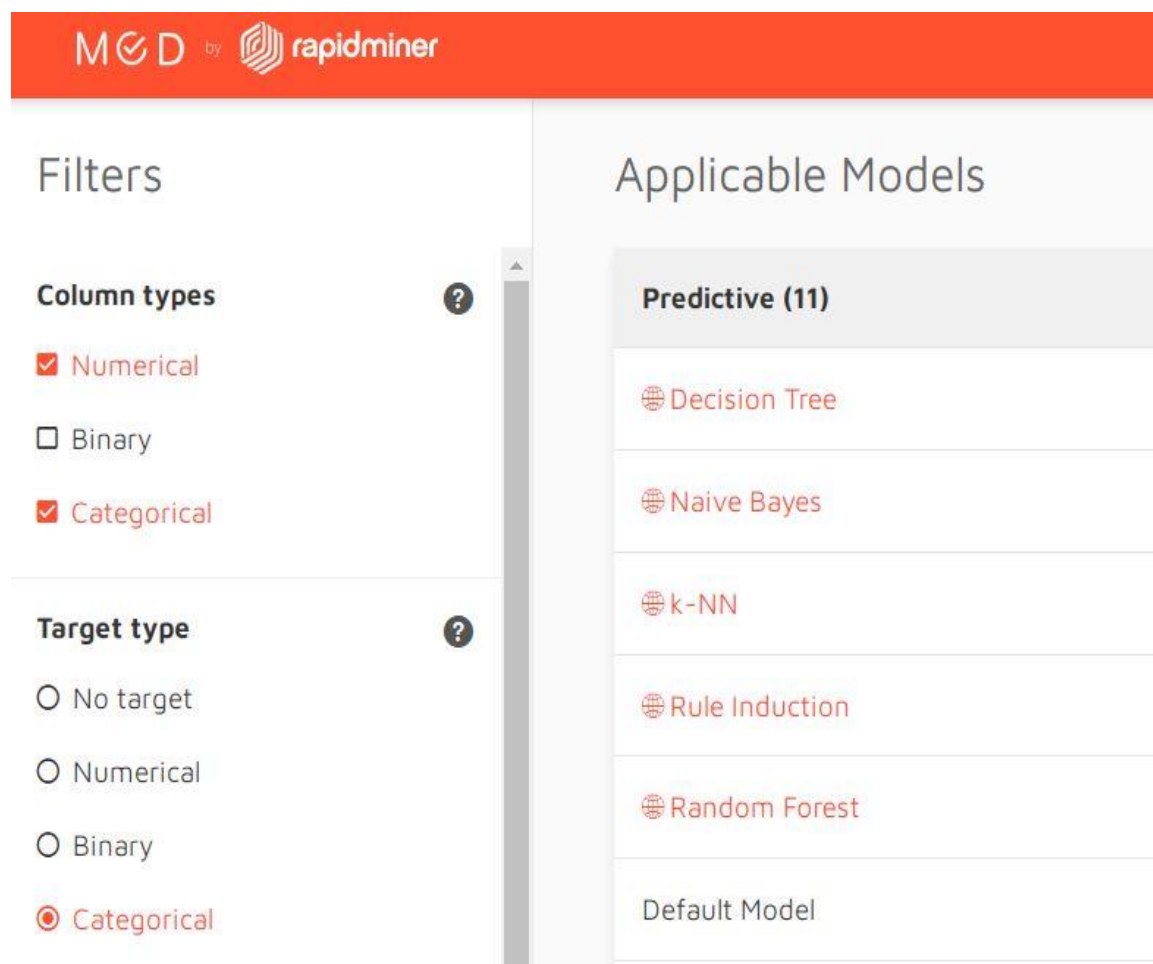


Figure 34



We also used the AutoModel function in RapidMiner on our prepared Training Set data to look at the recommended selection of models, their relative accuracy, along with other information such as recommended feature sets, and correlations.

AutoModel

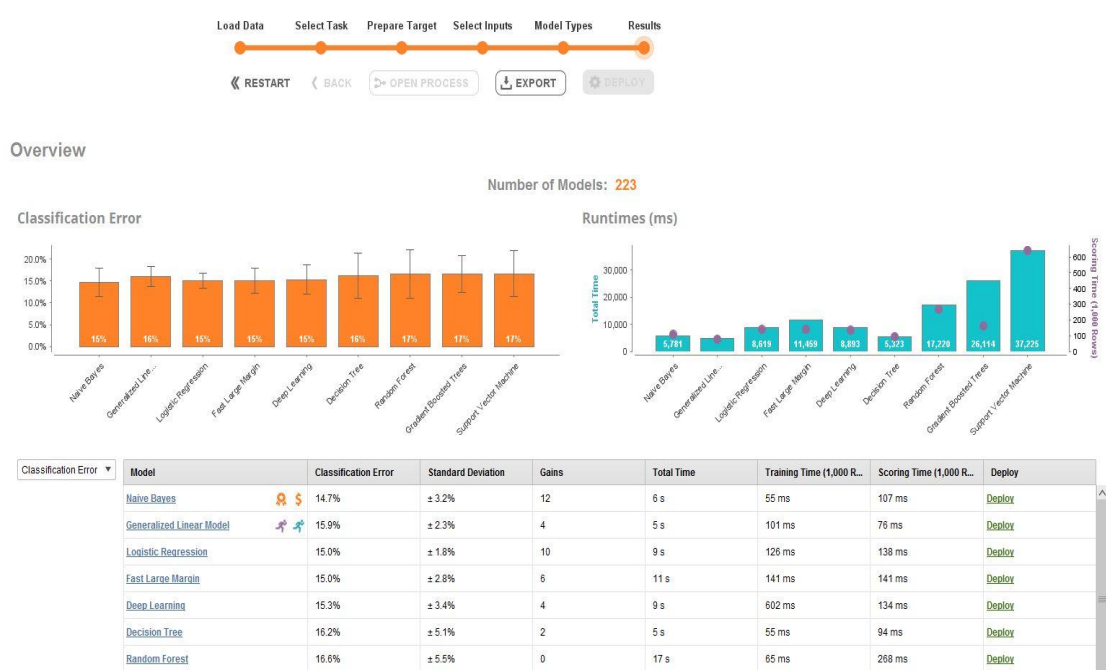


Figure 35

The AutoModel output re-confirmed that we would focus our Modelling Phase to look at the recommended list of five algorithms, as listed in the RapidMiner website screenshot above. K-NN and Rule Induction did not feature in the AutoModel output but we still choose to go with the recommendations from the RapidMiner web application.

The screen shot above is based on the very first AutoModel run on the original Kaggle dataset and reflects a linear regression approach, before we added the categorical feature to represent 'Quality_Score' bands. We ran AutoModel an additional number of time, with various stages of the Wine Quality dataset, to assess possible model algorithm selections.

The Wine data quality has already been determined to be very good, based on our analysis and work described in Section 3 and Section 4 of this document. Therefore it was expected that our modelling techniques with the above algorithms would not encounter any basic processing issues with the data structure or format.



5.2 Generate Test Design

Before building an actual model for deployment, it is general practice in this task within the Modelling Phase to generate a procedure to test the model's quality and validity.

In a supervised data mining task, such as the classification of wine quality in this project, it would be common to use error rates as quality measures for the models.

Training and Test Sets

We therefore separated our Wine Quality dataset into Training and Test sets.

The Training Set is used to build the model, and the quality of the model is subsequently estimated using the separate Test Set.

We build a Data Preparation Process in RapidMiner to prepare the Wine Quality csv data for use in Modelling. The final steps in that process were to split the overall dataset in a 70/30 ratio into the Training and Testing sets respectively.

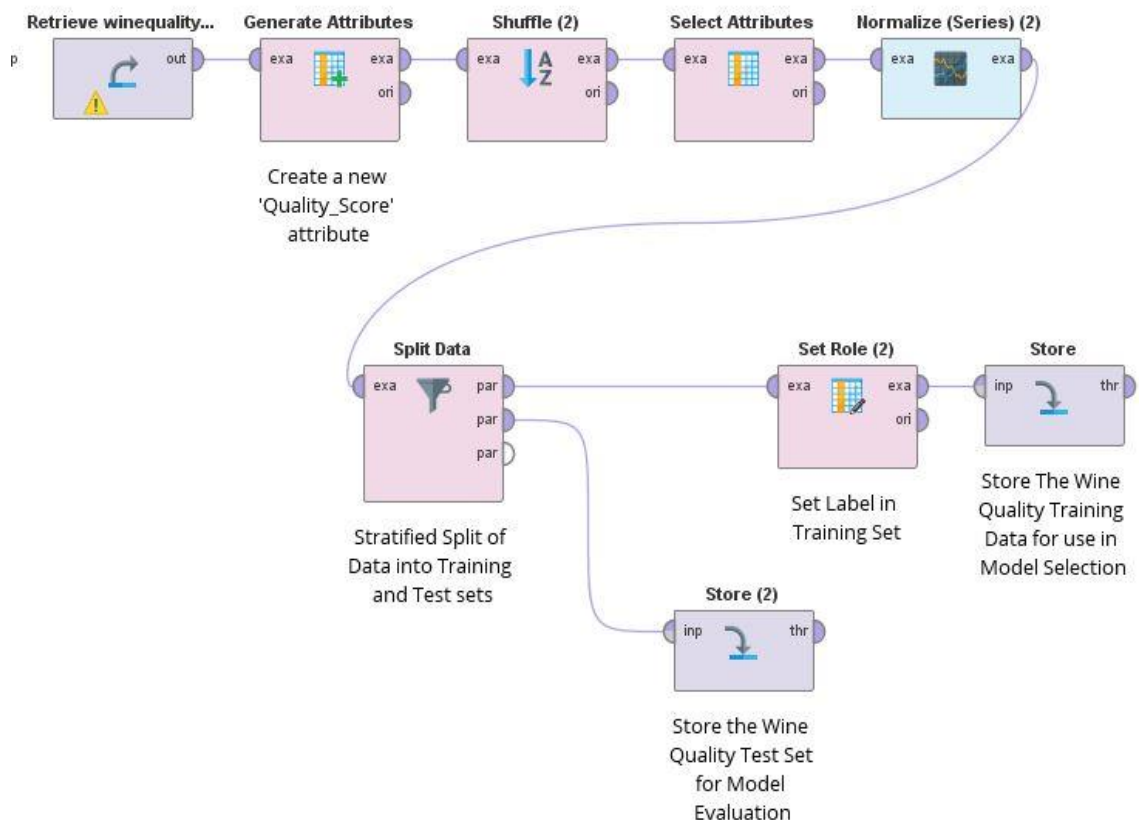


Figure 36

Storing the Training and Test Sets at the end of this process allows us to investigate the data stores independently before we enter the Modelling Phase.

The Modelling Phase processes in RapidMiner begin with the Training and Test Sets as their initial inputs.



5.3 Build Model

The purpose of this task is to run the modelling tool (RapidMiner) on the prepared dataset (Wine Quality) to create one or more models.

In our project, we choose to focus initially on five models and then focused on the one that was more consistently generating more accurate results with the Test Set data.

Cross Validation

To recap, for training our models we initially split the model into two sections which are '**Training data**' and '**Testing data**'.

In our RapidMiner modelling process we incorporated a Cross Validation operator, which dynamically splits the Training Set into **Validation Sets**.

Our modelling technique is to train the classifier using '**training data set**', tune the parameters using '**validation set**' and then test the performance of your classifier on unseen '**test data set**'. An important point to note is that during the training of the classifier only the training and/or validation set is available. The test data set is not used during the training of our Wine Quality classifier. The test set will only be available during testing the classifier.

To elaborate:

Training set: The training set is the material through which the computer learns how to process information. In our Wine Quality Data Preparation Phase we generated this Training Set so that our Machine Learning approach will use algorithms to perform the training part for a predictive model.

Validation set: Cross-validation is primarily used in applied machine learning to estimate the skill of a machine learning model on unseen data. A set of unseen data is used from the training data to tune the parameters of a classifier.

Test set: A set of unseen data used only to assess the performance of a fully-specified classifier. The Test Set for the Wine Quality model building task is one of the outputs from our Data Preparation Phase.

The layout of the RapidMiner process, with Cross Validation can be seen in this image.

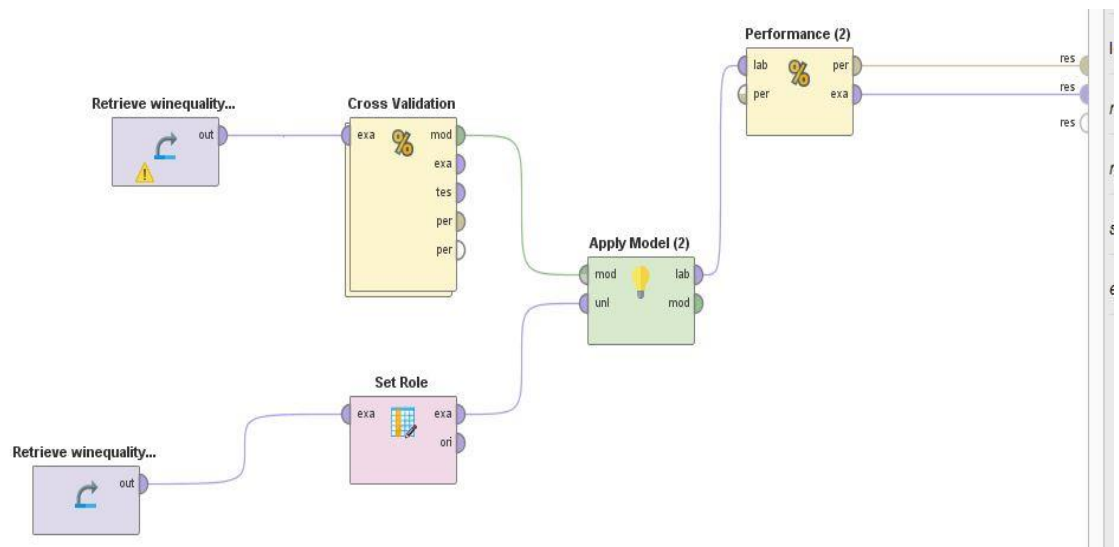


Figure 37

Drilling into the Cross Validation operator we can see an example of the following set up in RapidMiner (with Decision Tree);

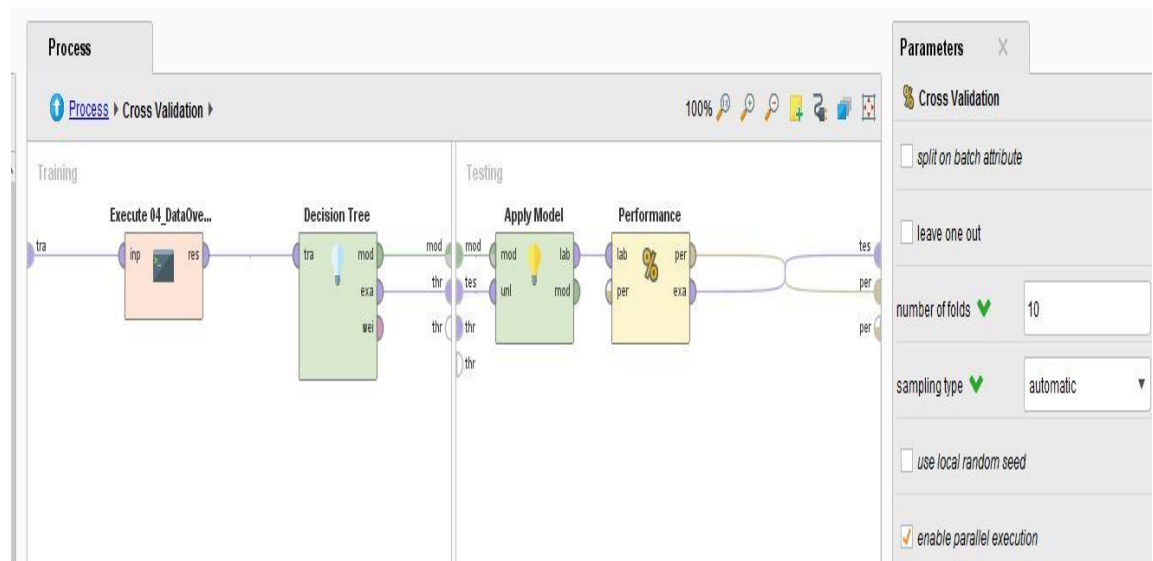


Figure 38

The left-hand side of the Cross Validation process runs multiple iterations using different slices of 'Training' and 'Validation' data.

Our Cross Validation operator uses a 10-fold parameter. For simplicity, we have included below a diagram that represents and illustrates the same process but for a 5-Fold Cross Validation split.

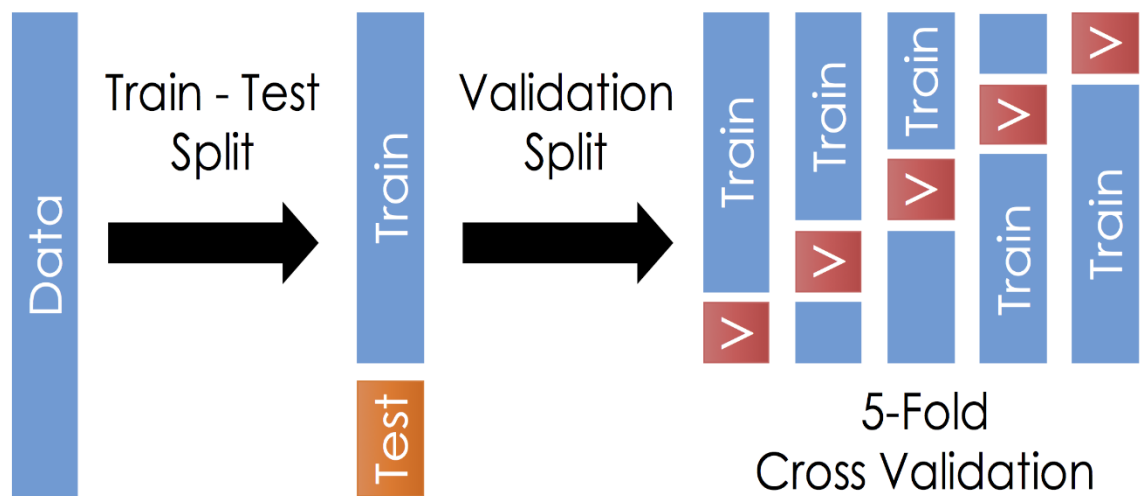


Figure 39

Once the Wine Quality data is divided into the 3 given segments in our RapidMiner processes, we could commence the training process.

RapidMiner provides operators to ensure the model is applied to the Training/Validation data using the Apply Model Operator. The Performance operator is used to evaluate how accurate the aggregate of the various validations were across the data 'splits'.

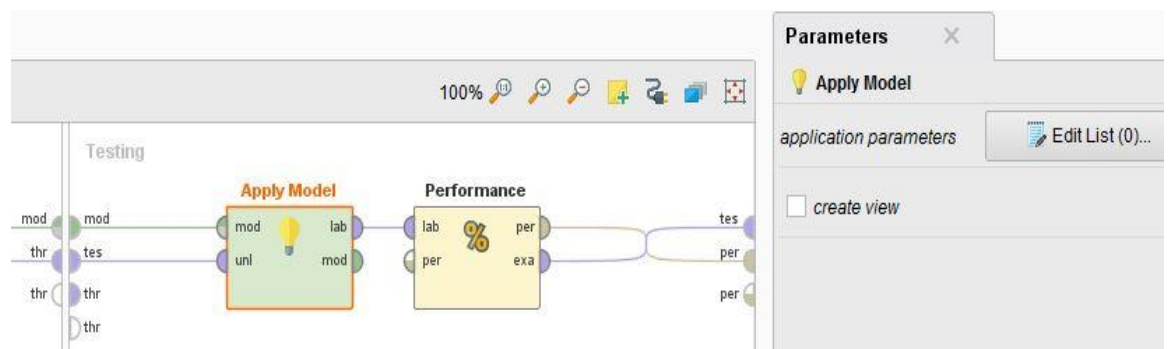


Figure 40

The output of the Performance operator in the above screenshots is related to the performance of the model on the Training data. Or primary interest is in the performance on the 'unseen' Wine Quality Test data, which is discussed later in this section of the document

Embedded Over and UnderSampling

In Section 4 of this report we described the challenge with the poor balance of data in our Wine Quality dataset. A sequence of Downsampling and Upsampling operators in RapidMiner were employed to attempt to generate a more balanced dataset for modelling.

Although those operators were described in the Data Preparation Phase, the actual implementation of the balancing routines is executed during the Cross Validation task.

We created a standalone RapidMiner process to perform the down sampling of 'Medium' wines, followed by the sequence of SMOTE operators to increase the rows of data for 'Poor' and 'High' quality wines

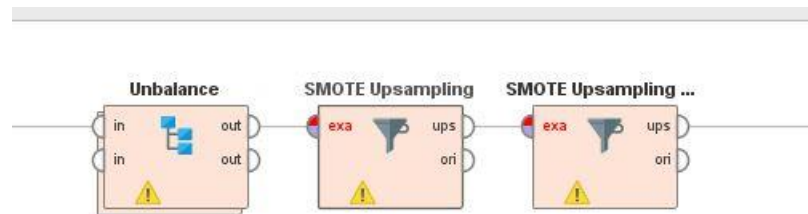


Figure 41

This routine is then invoked as the first step in the operators nested into the Cross Validation operator.



Why execute the balancing routine within the validation process?

Online resources recommended against executing the balancing operators just once before the Cross Validation. This would introduce a risk of 'over fitting' the data during the modelling process, by aligning the model too closely to a single set of synthetic data (and down sampled 'Medium' wines).

Thus for each of the 10-fold validations in our Cross Validation process there is a separate exercise to create new artificial data and remove certain 'Medium' rows.

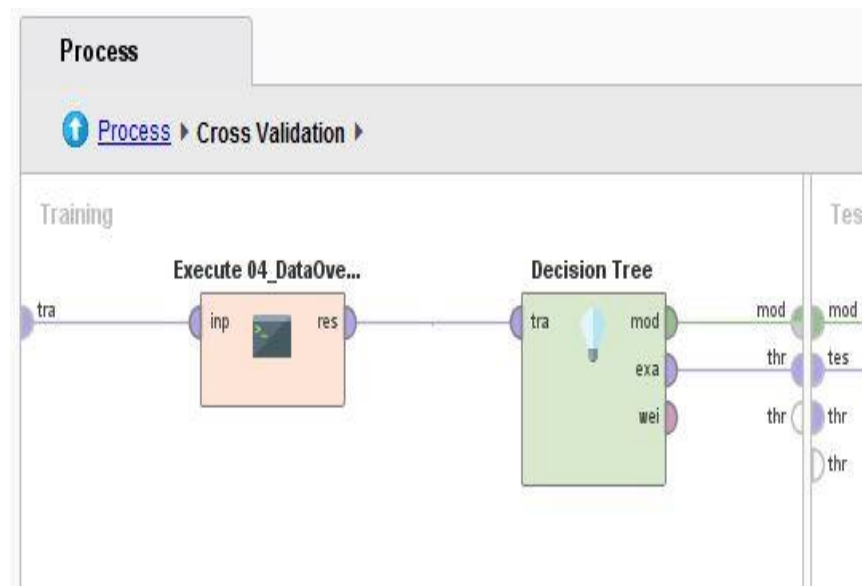


Figure 42



Models

Taking our lead from the AutoModel and RapidMiner web application recommendations we focused on building models with the following algorithms;

- Decision Trees
- K-NN
- Naive Bayes
- Rule Induction
- Random Forest

We choose to build separate models in RapidMiner rather than just use the output from AutoModel. We felt we needed to have as close as understanding and control of the overall Data Preparation Phase as possible, particularly the balancing challenges with the Wine Quality dataset.

The AutoModel performance output, based on both the original Kaggle dataset and the post Data Preparation Phase data, did not show a significant difference in accuracy between the various models.

We build a separate process for each of the five models in RapidMiner (and some other test processes);

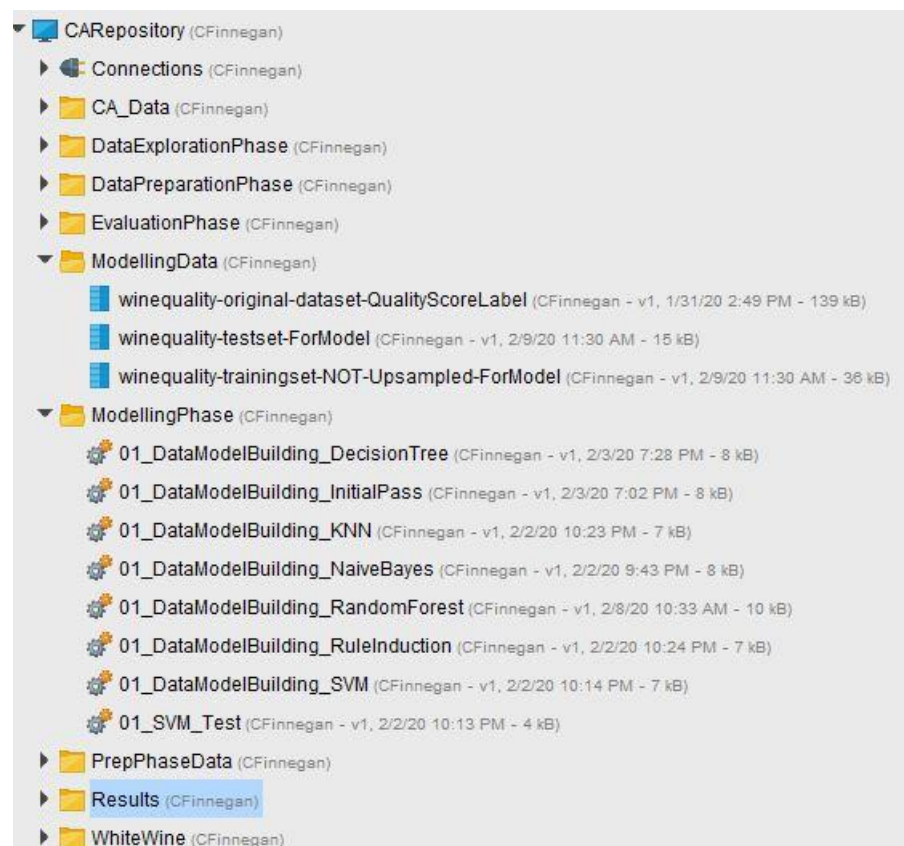


Figure 43



Again, there was no immediate standout model, although Decision Trees and Random Forest proved to be generally better with their spread of predictions.

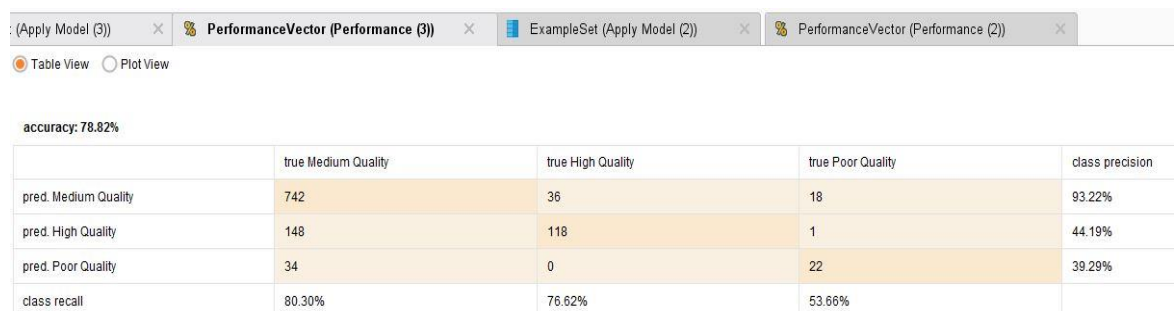
It has to be noted at this point that despite the multiple models we built and tested, no one was really acceptable. Our analysis started to move into finding the model that was 'least bad'.

For the remainder of the project we worked on the following steps;

1. Refining our balancing of the dataset and applying it to the Decision Tree model to determine the optimum settings for new artificial and down sampled data.
2. Refining the Random Forest model with the balanced dataset to produce the best possible results.

Tracking Performance – Training and Test Sets

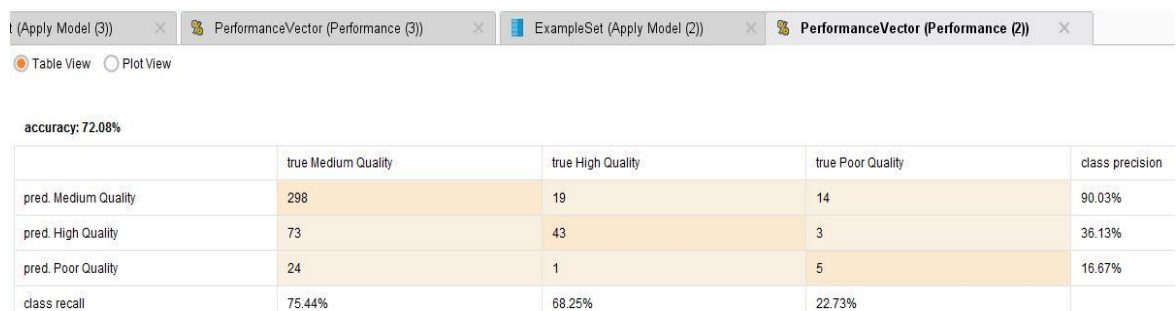
Looking at the **Decision Tree** Model we were able to produce a Training Set output with the following accuracy.



	true Medium Quality	true High Quality	true Poor Quality	class precision
pred. Medium Quality	742	36	18	93.22%
pred. High Quality	148	118	1	44.19%
pred. Poor Quality	34	0	22	39.29%
class recall	80.30%	76.62%	53.66%	

Figure 44

When applied to the 'unseen' data, in the Test Set, the model produced the following results.



	true Medium Quality	true High Quality	true Poor Quality	class precision
pred. Medium Quality	298	19	14	90.03%
pred. High Quality	73	43	3	36.13%
pred. Poor Quality	24	1	5	16.67%
class recall	75.44%	68.25%	22.73%	

Figure 45

Training Set outputs were not that impressive, but the Test Set performed even worse.



We felt that we should now focus on the ensemble algorithm approach with **Random Forest**.

Looking at the Random Forest Model we were able to produce a Training Set output with the following accuracy.

	true Medium Quality	true High Quality	true Poor Quality	class precision
pred. Medium Quality	796	27	15	94.99%
pred. High Quality	104	127	0	54.98%
pred. Poor Quality	24	0	26	52.00%
class recall	86.15%	82.47%	63.41%	

Figure 46

When applied to the 'unseen' data, in the Test Set, the model produced the following results.

	true Medium Quality	true High Quality	true Poor Quality	class precision
pred. Medium Quality	318	16	15	91.12%
pred. High Quality	57	46	2	43.81%
pred. Poor Quality	20	1	5	19.23%
class recall	80.51%	73.02%	22.73%	

Figure 47

Test Set performance with Random Forest was far from ideal, and only really produced acceptable results with 'Medium' quality wines. However, being guided by the Performance operator outputs from all the Models we built, this Random Forest outcome was the most 'accurate'.

Parameter Settings

Looking at our Random Forest set up within the Cross Validation operator in the RapidMiner process we applied an additional 'Bagging' operator.

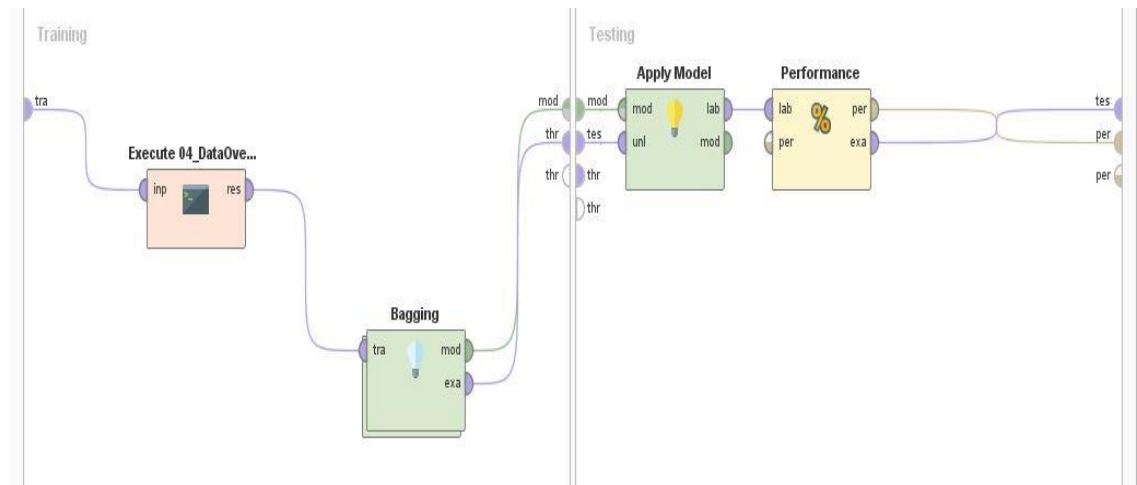


Figure 48

Double-clicking through the Bagging operator, we placed the Random Forest operator.

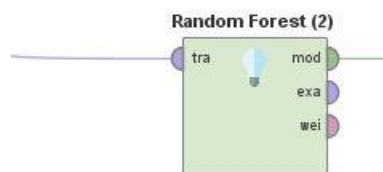


Figure 49

Although Random Forest is itself an ensemble algorithm using multiple Decision Trees to reach an aggregate result, we chose to run it within multiple (10) Bagging executions.

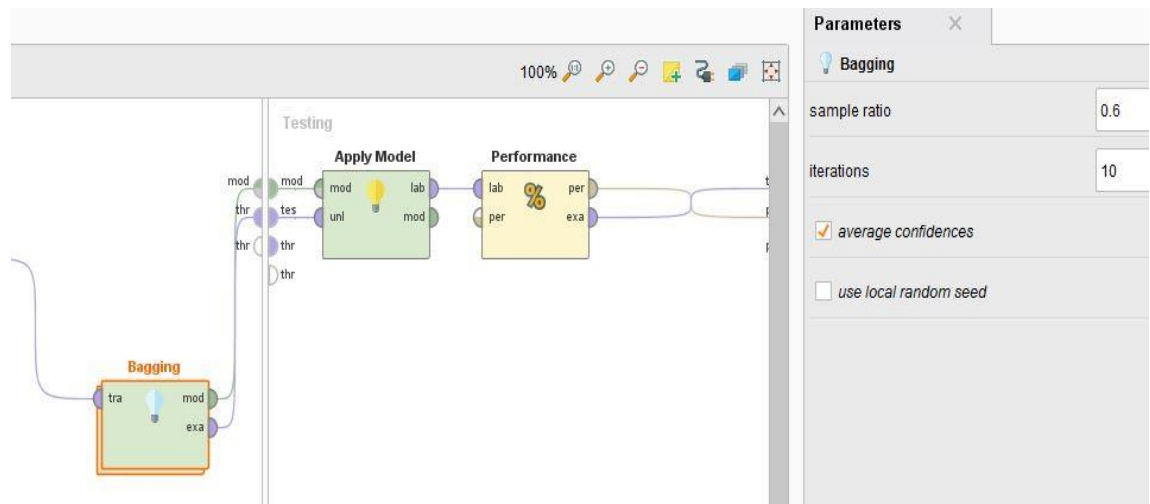


Figure 50

Although the feature set is not large, the bagging routine will select a 60% random sample of features (with replacement) into ten 'bags' and then execute the Random Forest operator. We wanted to attempt as broad a selection of executions to find the best model result.

Improvements in accuracy were delivered with this RapidMiner set up but only really in the order of 5%+ across the three wine categories.



5.4 Assess Model

In general, the data mining engineer will interpret the models according to his domain knowledge, the data mining success criteria, and the desired test design.

Business Domain Criteria

In our business domain for this project, we want a predictive model that will help employees provide a high level recommendation on wine quality based on its known chemical composition (as opposed to actually sampling every new type of red wine purchased from the wholesaler!)

We judged the models in terms of how well they would predict the right quality category for a 'new' wine (using the Test set). The Performance operator provided us with details on how often the prediction was right, and if it was wrong what was the type of classification error.

Ranking Models with Confusion Matrices

We attempted to rank the models based on the outputs of the RapidMiner Performance operator when the model is applied to the Test Set.

The Random Forest model was by far the most computationally expensive but looking at the Confusion Matrices for each model, a sample of which can be seen in Section 5.3, it provided the 'least bad' set of predictions.

Model Assessments

Models are assessed based on evaluation criteria, looking for general accuracy but also taking business success criteria into account.

The model assessment task in the Modelling Phase delivered a model that was good at predicting if a wine quality was actually in the 'Medium' category'.

For 'High' quality wine, the model was very unlikely to mis-classify a new wine as 'Poor'. However, the model was often not much better than a coin toss in terms of predicting if a 'High' quality wine as actually a 'Medium'.

The Resultant Model of Choice for Wine Quality

The Random Forest model was our choose model, although it clearly comes with a number of key flaws.



Revised Parameter Settings

Looking back over the AutoModel results we looked at the recommended settings for a Random Forest model, when applied to our Wine Quality dataset.

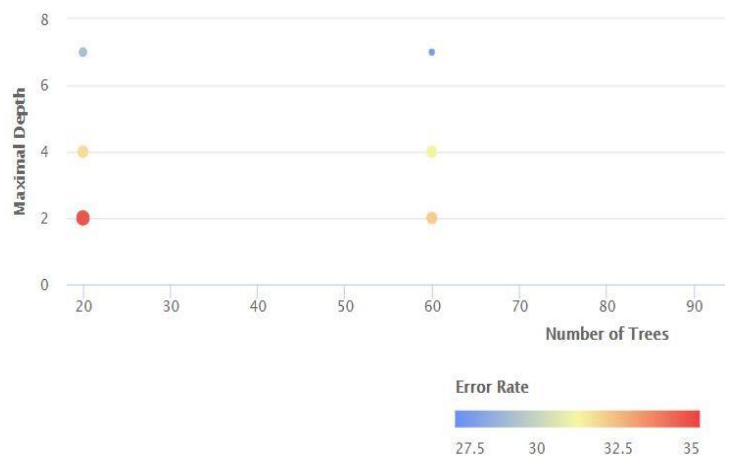


Random Forest - Optimal Parameters

Optimal Parameters

Number Of Trees: 100
Maximal Depth: 7

Error Rates for Parameters



Number of Trees	Maximal Depth	Error Rate
20	2	34.6%
60	2	32.1%
100	2	31.5%
140	2	32.4%

Figure 51

We updated the Random Forest operator with these settings – *Number of Trees* = 100 and *Maximal Depth* = 7 - but it made no perceptible impacts on the accuracy results.



6 Evaluation

6.1 Evaluate Results

Previous evaluation steps dealt with factors such as the accuracy and generality of the model. This task in the Evaluation Phase assesses the degree to which the model meets the business objectives and determines if there is some business reason why this model is deficient.

Assessment With Respect To Business Success Criteria

At best, we derived a model that will be successful enough for our business purposes to correctly predict a 'Medium' wine in most circumstances.

The question is obviously "would that be acceptable for the business"?

As the vast majority of wines are 'Medium' anyway, a predictive model which only really operates for that category is of limited use. The real business value is predicting a 'High' or 'Poor' quality wine as frequently as possible.

This project, with the express objective of building a predictive model for wine quality, has provided some interesting insights on the nature of the source data but has probably failed to deliver on the business needs.

6.2 Review Process

Have we overlooked any factors in our data mining process? Can we determine any flaw in our modelling approach?

Data Structure

The mixture of chemical components in wine that combine to produce a 'High' quality drink is probably a more complex relationship than 1600 rows can adequately capture.

It is our view that the relationship between the ingredients in wine needs much more data to allow a meaningful model to be created.

The challenge is therefore to reach out into the business world of wine makers and consumers and find further studies and data upon which we can re-attempt to build our predictive model.



6.3 Determine Next Steps

For this task in the Evaluation Phase the project needs to decide how to proceed.

Determining Next Steps

In the 'real world', our wine Quality predictive model would probably have too many gaps and deficiencies to actually roll out into a production setting.

However, for the purposes of the CA, and for academic curiosity at least, we will move to the Deployment Phase.



7 Deployment

7.1 Plan Deployment

This task takes the evaluation results and determines a strategy for deployment.

Deployment Strategy

Our analysis of the Wine Quality dataset produced a flawed model. However, if we still wished to proceed with a production rollout, these would be the likely steps;

- Step 1: Assume that our company has a RapidMiner server running across the company network.
- Step 2: Deploy our model onto the RapidMiner server and have it made available to key back office staff.
- Step 3: When new wine is being purchased an EXCEL template will be provided into which an employee enters the chemical details.
- Step 4: This EXCEL spreadsheet forms the input to the RapidMiner model, which incorporates the necessary data manipulation routines.
- Step 5: The RapidMiner server would output another EXCEL spreadsheet, in a prescribed format, with a clearly identified score for each wine.
- Step 6: This output spreadsheet would be shared with all employees as a reference sheet for the new in-store wine.

7.2 Plan Monitoring and Maintenance

Monitoring and maintenance are important issues if the data mining result becomes part of the day-to-day business and its environment.

Obtain More Data in Future and Re-Model

It may be possible to obtain new, independently verified, data on Wine Quality. Thus new models could be produced with these enlarged datasets to attempt to improve accuracy.

The new, more accurate, models would then be deployed via RapidMiner server when available.



7.3 Produce Final Report

At the end of the project, the project team writes up a final report.

This is that report.

7.4 Review Project

This is the task within the Evaluation Phase when assessments are made about what went right and what needs to be improved.

7.4.1 Conclusions

- **Data is 'king'.** We just did not have enough meaningful real world data to build an effective model that would cover all three categories of wine.
- **Even with artificial data the problems with building an effective model persisted.** The complexity of the interactions of the wine ingredients that produce the measure of 'quality' are probably much greater than we realised. It is likely we would need at least tens of thousands of data rows to generate a meaningful model.
- **Even a manual process, supplementing the predictive model outputs, would be an ineffective solution for our business needs.** Failing to predict a 'High' quality wine 50% of the time to customers could have a detrimental impact of business reputation.

7.4.2 Learnings

- **RapidMiner is a great tool for quickly modelling data and refining the process.** We had some experience in Python Machine Learning techniques and that allowed for quick low level investigations. However the RapidMiner UI and support documentation allowed us to progress quickly through the project objectives.
- It really does seem to us that, as the saying goes, that there really are no good models, just some useful ones (even if they are 'bad' and we learned what will not work).



8 Appendices and References

8.1 Appendix A: Understanding Wine and Types

Wine is an alcoholic beverage made from grapes which is fermented without the addition of sugars, acids, enzymes, water, or other nutrients

Red wine is made from dark red and black grapes. The colour usually ranges from various shades of red, brown and violet. This is produced with whole grapes including the skin which adds to the colour and flavour of red wines, giving it a rich flavour.

White wine is made from white grapes with no skins or seeds. The colour is usually straw-yellow, yellow-green, or yellow-gold. Most white wines have a light and fruity flavour as compared to richer red wines.

Understanding Wine Attributes and Properties

- **Fixed acidity:** Acids are one of the fundamental properties of wine and contribute greatly to the taste of the wine. Reducing acids significantly might lead to wines tasting flat. Fixed acids include tartaric, malic, citric, and succinic acids which are found in grapes (except succinic).
- **Volatile acidity:** These acids are to be distilled out from the wine before completing the production process. It is primarily constituted of acetic acid though other acids like lactic, formic and butyric acids might also be present. Excess of volatile acids are undesirable and lead to unpleasant flavour. In the US, the legal limits of volatile acidity are 1.2 g/L for red table wine and 1.1 g/L for white table wine.
- **Citric acid:** This is one of the fixed acids which give a wine its freshness. Usually most of it is consumed during the fermentation process and sometimes it is added separately to give the wine more freshness.
- **Residual sugar:** This typically refers to the natural sugar from grapes which remains after the fermentation process stops, or is stopped.
- **Chlorides:** This is usually a major contributor to saltiness in wine.
- **Free sulphur dioxide:** This is the part of the sulphur dioxide that when added to a wine is said to be free after the remaining part binds. Winemakers will always try to get the highest proportion of free sulphur to bind. They are also known as sulphites and too much of it are undesirable and give a pungent odour.
- **Total sulphur dioxide:** This is the sum total of the bound and the free sulphur dioxide. This is mainly added to kill harmful bacteria and preserve quality and freshness. There are usually legal limits for sulphur levels in wines and excess of it can even kill good yeast and give out undesirable odour.



- **Density:** This can be represented as a comparison of the weight of a specific volume of wine to an equivalent volume of water. It is generally used as a measure of the conversion of sugar to alcohol.
- **pH:** Also known as the potential of hydrogen, this is a numeric scale to specify the acidity or basicity the wine. Fixed acidity contributes the most towards the pH of wines. You might know, solutions with a pH less than 7 are acidic, while solutions with a pH greater than 7 are basic. With a pH of 7, pure water is neutral. Most wines have a pH between 2.9 and 3.9 and are therefore acidic.
- **Sulphates:** These are mineral salts containing sulphur. Sulphates are to wine as gluten is to food. They are a regular part of the winemaking around the world and are considered essential. They are connected to the fermentation process and affects the wine aroma and flavour.
- **Alcohol:** Wine is an alcoholic beverage. Alcohol is formed as a result of yeast converting sugar during the fermentation process. The percentage of alcohol can vary from wine to wine. Hence it is not a surprise for this attribute to be a part of this dataset. It's usually measured in % vol or alcohol by volume (ABV).
- **Quality:** Wine experts graded the wine quality between 0 (very bad) and 10 (very excellent). The eventual quality score is the median of at least three evaluations made by the same wine experts.
- **wine_type:** Since we originally had two datasets for red and white wine, we introduced this attribute in the final merged dataset which indicates the type of wine for each data point. A wine can either be a 'red' or a 'white' wine. One of the predictive models we will build in this chapter would be such that we can predict the type of wine by looking at other wine attributes.
- **quality_label:** This is a derived attribute from the quality attribute. We bucket or group wine quality scores into three qualitative buckets namely low, medium and high. Wines with a quality score of 3, 4 & 5 are low quality, scores of 6 & 7 are medium quality and scores of 8 & 9 are high quality wines. We will also build another model in this chapter to predict this wine quality label based on other wine attributes.



8.2 Appendix B: The White Wine Dataset

The limitations with lack of data are evident in this project.

We found a dataset with the same structure as our original red wine data, but for white wine instead.

As an academic exercise we created a subset of processes with their own CRISP-DM approach to incorporate the white wine data and determine if it improved the results from our Random Forest model.

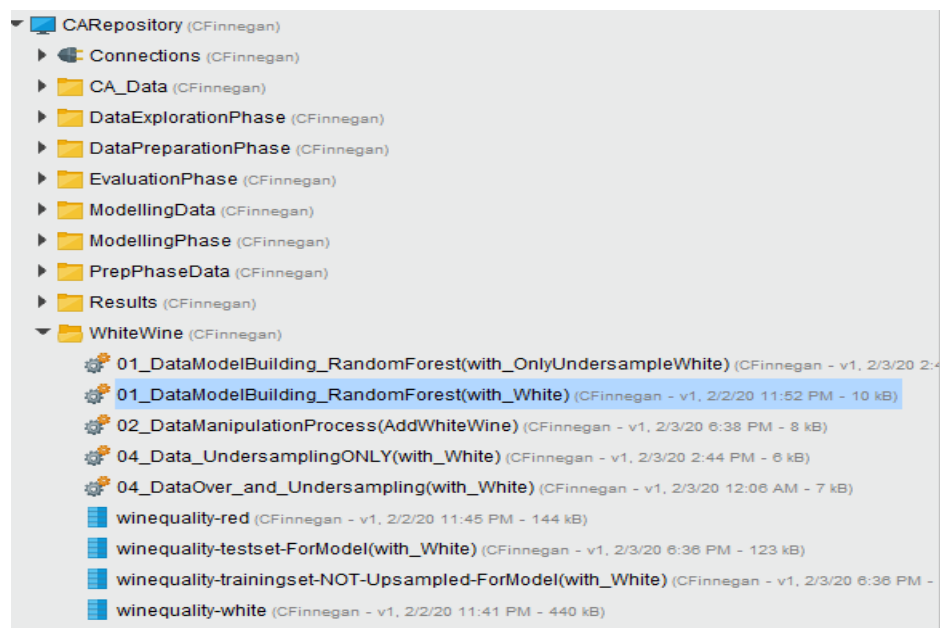


Figure 52

Adding the white wine data increased the overall dataset size but there were still relatively few 'Poor' wine examples.

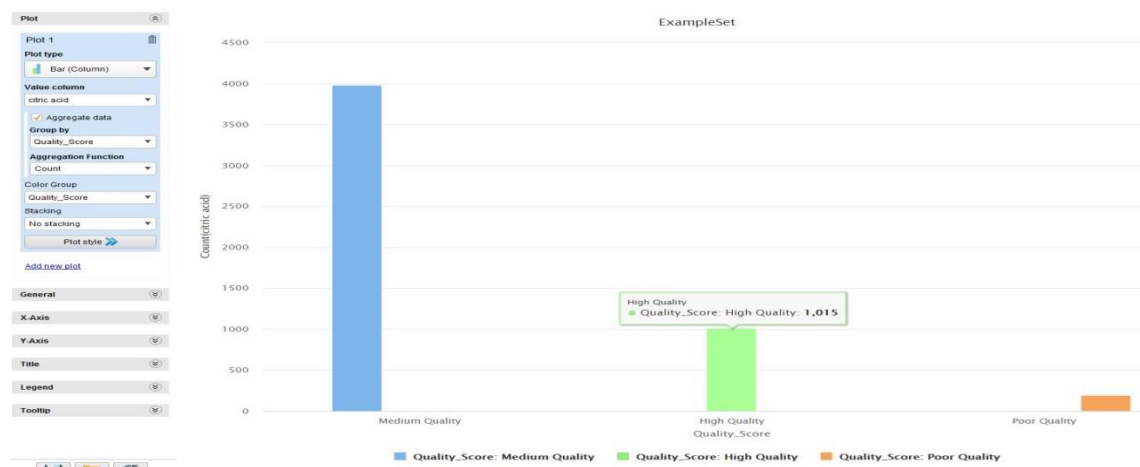


Figure 53



The Data Manipulation Process was extended to read in and merge the white wine date with the original red wine dataset.

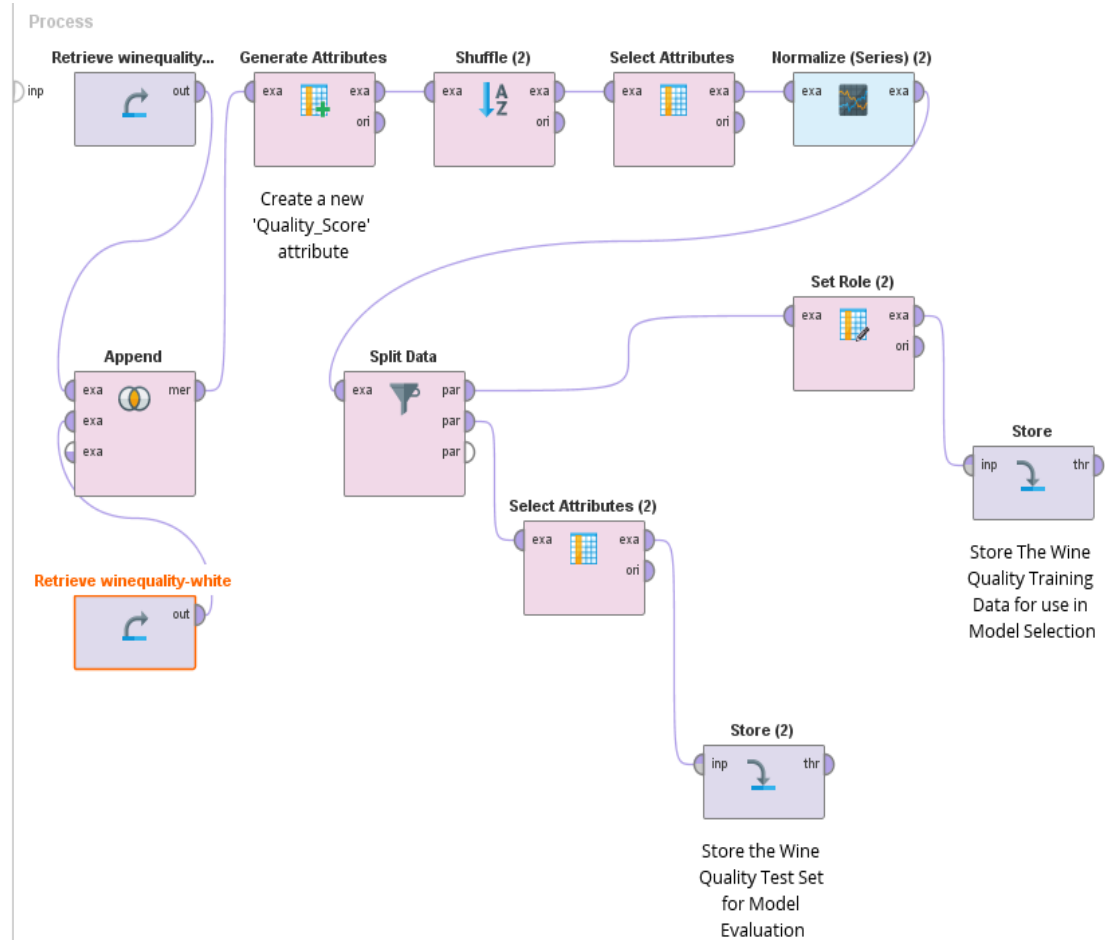


Figure 54

The same Random Forest Model was applied.

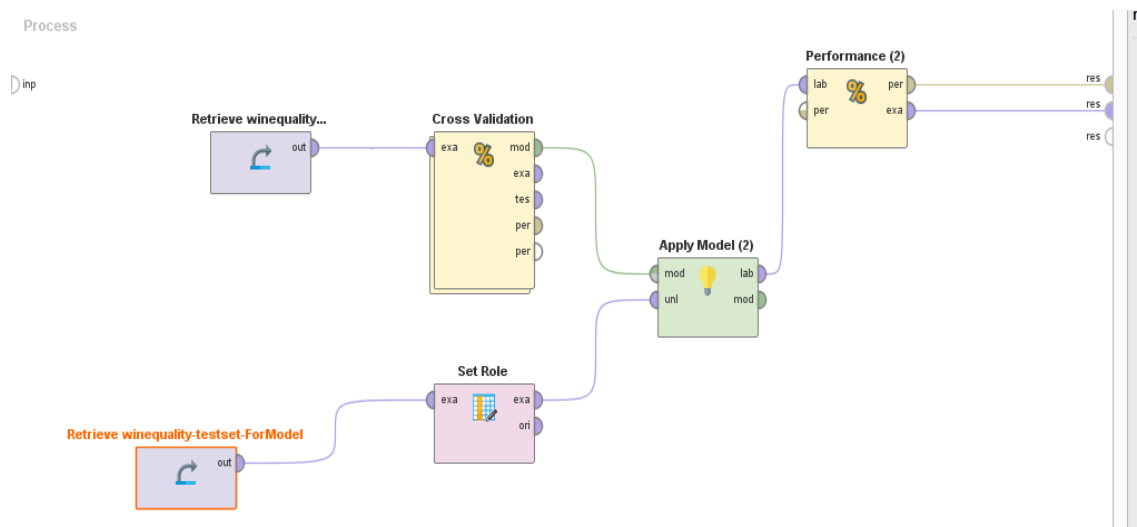


Figure 55



However, the end results from the model did not show any significant improvement in accuracy.

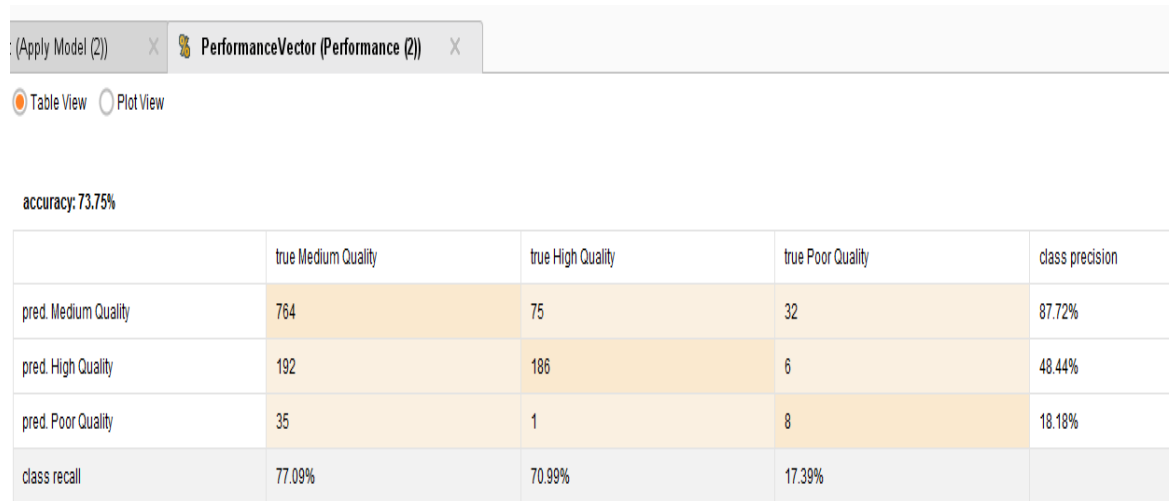


Figure 56

In our view, the wine dataset would have to be at least an order of magnitude bigger, or even larger, in order to provide a meaningful predictive model for wine quality.



8.3 Appendix C: References

The use of this Wine Quality dataset in this CA acknowledges the source publication:

P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modelling wine preferences by data mining from physicochemical properties. In Decision Support Systems, Elsevier, 47(4):547-553, 2009.